NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

FAR FIELD EXTRAPOLATION TECHNIQUE USING CHIEF ENCLOSING SPHERE DEDUCED PRESSURES AND VELOCITIES

By

Robert M. Drake
December 2003

NUWC Thesis Advisor: S.E. Forsythe
NPS Thesis Advisor: S.R. Baker

Approved for Public Release; Distribution is Unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** December 2003	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE: Far field extrapolation technique using CHIEF enclosing sphere deduced pressures and velocities			**5. FUNDING NUMBERS**
6. AUTHOR(S) Robert M. Drake			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

A Combined Helmholtz Integral Equation Formulation (CHIEF)-defined enclosing sphere placed around an acoustic projector is used to calculate far field response data from near field measurements. Pressure response data at a specified frequency is obtained from a near field linear array. Helmholtz integral relations for the enclosing sphere and integrals of the free-space Green's function and its gradient for defined near field point locations are used along with physical assumptions to form an overdetermined system. The overdetermined system is solved via least squares yielding values of pressure and velocity corresponding to defined locations on the enclosing sphere. The enclosing sphere's values of pressures and velocities are then used with integrals of the free-space Green's function and its gradient to calculate far field response.

14. SUBJECT TERMS Near Field, Far Field, CHIEF, Least Squares, Combined Helmholtz Integral Equation Formulation, USRD			**15. NUMBER OF PAGES** 191
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release, distribution is unlimited

FAR FIELD EXTRAPOLATION TECHNIQUE USING CHIEF ENCLOSING SPHERE DEDUCED PRESSURES AND VELOCITIES

Robert M. Drake
Naval Undersea Warfare Center Division Newport
B.S.E., University of Central Florida, 1983

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ENGINEERING ACOUSTICS

from the

NAVAL POSTGRADUATE SCHOOL
December 2003

Author: Robert M. Drake

Approved by: S.E. Forsythe
Thesis Advisor

S.R. Baker
Thesis Advisor

Kevin B. Smith, Chairman
Engineering Acoustics Academic Committee

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

A Combined Helmholtz Integral Equation Formulation (CHIEF)-defined enclosing sphere placed around an acoustic projector is used to calculate far field response data from near field measurements. Pressure response data at a specified frequency is obtained from a near field linear array. Helmholtz integral relations for the enclosing sphere and integrals of the free-space Green's function and its gradient for defined near field point locations are used along with physical assumptions to form an overdetermined system. The overdetermined system is solved via least squares yielding values of pressure and velocity corresponding to defined locations on the enclosing sphere. The enclosing sphere's values of pressures and velocities are then used with integrals of the free-space Green's function and its gradient to calculate far field response.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGEMENTS

This work represents the culmination of the collaborative Master of Science Engineering Acoustics distance learning program between the Naval Postgraduate School and the Naval Undersea Warfare Center Division Newport (2000 – 2003). A special acknowledgement is made to Mr. Stephen Forsythe, NUWCDIVNPT, for assistance in defining a program of study and approach leading to the body of work presented herein.

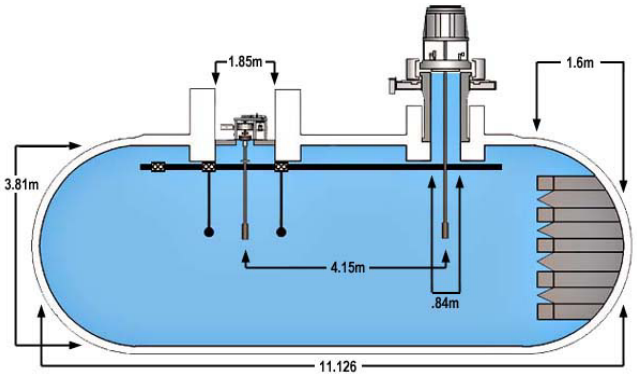
THIS PAGE INTENTIONALLY LEFT BLANK

I. BACKGROUND

The testing and characterization of U.S. Navy underwater acoustic devices remains a fundamentally important activity linked directly to both fleet and mission readiness. Testing of new prototypes, sampling of production line devices and re-qualification of in-service assets are all components of this readiness. These testing efforts represent measures put in place to help assure that the required level of acoustic performance is being achieved.

The Underwater Sound Reference Division (USRD) represents the Navy's primary activity for underwater acoustic calibration, test, and evaluation measurements. In this capacity, the USRD maintains specialized measurement facilities with the capability to simulate real-world ocean environments. The USRD is part of the Submarine Sonar Department of the Naval Undersea Warfare Center Division Newport located in Newport, Rhode Island (www.npt.nuwc.navy.mil/USRD).[1]

Within the USRD is an acoustic measurement facility designated as the Acoustic Pressure Tank Facility (APTF). The APTF test vessel consists of a closed steel tank 3.81 meters in diameter and 11.1 meters in length with two access ports. The facility allows the acoustic test and evaluation of devices and materials over a wide range of simulated ocean environments. Temperatures from 2 degrees Celsius to 35 degrees Celsius and hydrostatic pressures up to 18.6 megapascal (a depth equivalent of 1860 meters) can be achieved.



Naval Undersea Warfare Center Division Newport - Acoustic Pressure Tank Facility

The APTF represents a significant U.S. Navy capital investment. As Navy demands and needs have changed over the years, it has become increasingly more important to revitalize existing Navy assets to meet new requirements. While it is a common need to characterize the performance of acoustic devices under realistic ocean environmental conditions, in some instances these devices may require test distances in excess of that achievable within the confines of existing Navy test vessels. As an example, these devices may entail designs that have large apertures yet may also function at relatively high frequencies (as the U.S. Navy High Frequency Sail Array). This distance required for proper acoustic characterization is designated as that distance where far field conditions [3] occur. Test distances that meet or exceed this far field criterion allow the device to function as it would in the free-field condition of the open ocean. Testing at distances that fall short of this criterion are designated as being tested in the near field. Testing in the near field is subject to contaminations that obscure the true far field performance of the device and thus is typically not desirable.

This thesis effort addresses a technique that uses near field acoustic data with a suitable math model to extrapolate far field performance. The use of "CHIEF for MATLAB"[2] coupled with a least squares solution [7] methodology are utilized in this technique. The data output rendered from the technique mimics data expected within test vessels that would be prohibitively expensive to fabricate and maintain.

The remainder of this thesis is organized as follows:

Chapter II addresses the definition and meaning of near-field and far- field conditions.

Chapter III provides an overview of the Combined Helmholtz Integral Equation Formulation (CHIEF) and the **A** and **B** matrices that are used in subsequent computations in the thesis methodology.

Chapter IV details the steps of the thesis near-field to far-field methodology and criteria for assessing the data output.

Chapter V provides an example solution set using the methodology.

Chapter VI provides an error analysis of the technique when positional uncertainties of the various acoustic devices used during a test exist.

Chapter VII looks at other specific test examples inclusive of testing in more extreme near-field cases and test results expected from what is likely to be considered as a typical test set-up and scenario.

THIS PAGE INTENTIONALLY LEFT BLANK

II. NEAR-FIELD/FAR-FIELD OVERVIEW

A. INTRODUCTION

A fundamental physical dimension addressed in conducting acoustic measurements is separation distance. The desired separation distance between devices in a test setup places them at distances where 1/r spreading occurs (with r defined as that distance between the projecting and receiving devices). This mimics expected performance in a free-field state and enables an accurate characterization of the device for the typical ocean environment for which it is designed. 1/r spreading gives the expected 6 dB loss when separation distance is doubled [3]. This type of spreading occurs when "far-field" conditions have been met.

Testing of a transducer with finite dimensions (i.e., not a point source) at distances inside the far-field limit results in performance characterization that is obscured by phase superposition of the comprising elements. That is, the complex response measured at a particular point is made up of a number of individual responses each with widely varying phase terms. Thus, the "near field" of a particular set-up is characterized by radical fluctuations in pressure with changes in range (in a given direction). This obscures the nature of the device's true far field performance.

The following curve demonstrates the variation in the near field and far field responses seen for a particular acoustic device (a linear array). The response curve was obtained by summing up the complex responses from all source elements at each of several specified ranges. Note the widely changing response in the near field of the device and the 1/r spreading resulting at the farther ranges.

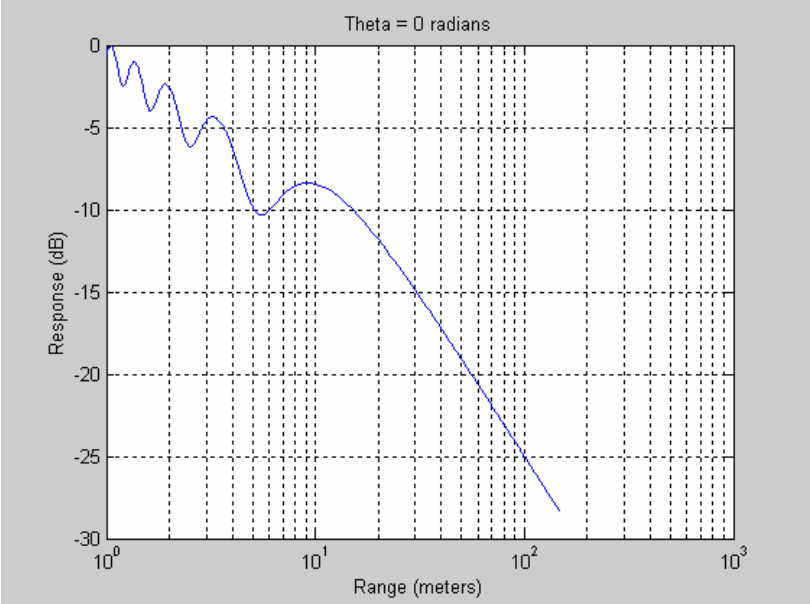


Figure 1 – Pressure Response with Change in Range

The field data at varying ranges in the previous figure (Figure 1) was computed for a 30.5 cm linear array using a test frequency of 600 kHz. A far field distance of approximately 37 meters is estimated for this device. The data set was normalized to the maximum response computed. Computations were based upon modeling the source as a "Continuous Line Source" as detailed in Kinsler and Frey, Section 7.3 (4th Edition) [10].

B. THEORY

The properties of the corresponding acoustic field for a given device can typically be estimated from the geometry of the source and its wavelength, λ, of operation. The extent of the near field can be estimated through the examination of the acoustic contributions from the device's various geometric components. The figure that follows is used to examine these details.

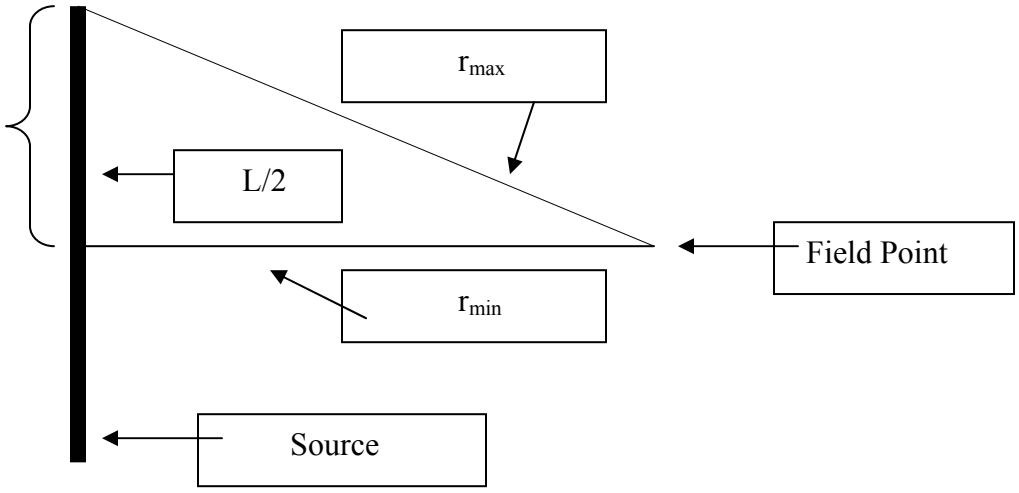


Figure 2 – Geometry used in estimating extent of Near Field

In the above figure, r_{max}, represents the distance from the furthest element of the source (in this example this is a line array of length L) to a field point. The magnitude, r_{min}, represents the distance from the nearest element of the source to the same field point. The difference between these two magnitudes is defined as $\Delta r = r_{max} - r_{min}$. At great distances, this difference (Δr) is minimal. At closer distances and as the field point approaches the source on axis, this difference can be significant. When the difference approaches $\lambda/2$ (over the difference obtained at great range), the phases of the signals from the individual points on the source combining at the field point will be of sufficiently different value. The net effect of this is that the axial pressure, $P_{ax,}$ can change radically over small changes in distance (see Figure 1).[4,10]

Referencing Figure 2 again gives the following expression for indicating a reasonable boundary for expected far field performance:

$$r_{min} / L \approx L / 4\lambda \qquad (1)$$

L/2 represents half of the maximum dimension of the source.

A rule of thumb for far field separation distance that has been used in the USRD facilities has been based upon

$$r > \frac{D^2}{\lambda}, \tag{2}$$

with D equal to the maximum dimension of the aperture [5], [9].

III. CHIEF - INTRODUCTION

A. BACKGROUND

The CHIEF (Combined Helmholtz Integral Equation Formulation) program was originally developed at the Naval Ocean Systems Center (NOSC) from 1965-1970 by Drs. H.A. Schenck and G.W. Benthien. It was originally developed to compute acoustic radiation that would be produced from a given arbitrarily defined body. Over the years its capabilities have been expanded by others to utilize more powerful and available computing hardware and to include plane-wave scattering and finite-element defined input descriptions [6] .

The version of CHIEF used in this effort was written for implementation with MATLAB by Mr. Stephen Forsythe, NUWCDIVNPT, during fiscal years 1999-2000. This represents a unique implementation of CHIEF that takes advantage of the matrix handling capabilities of a commercial off the shelf product. While achieving the functionality of the original CHIEF program, Mr. Forsythe further augmented this version to utilize spatially defined material properties rather than just relying on boundary value problem descriptors [2] . The open nature of the coding that was implemented also allows access to non-traditional CHIEF outputs. It is this latter implementation detail that made this version of CHIEF the proper choice for this effort. Specifically, the extraction of the CHIEF A and B matrices for use outside of the protocols of CHIEF was the unique feature that this version provided and that this thesis methodology required.

B. CHIEF THEORY

This section provides a brief glimpse at the definition and formulation of the CHIEF **A** and **B** matrices and the use of the Helmholtz Integral formula to deduce pressure at a point in space. This thesis effort does not concentrate on the exacting derivation of these parameters but rather provides an overview. More complete details

may be found on this derivation in the documentation provided with the CHIEF 1988 release [6] .

For both acoustic radiation and acoustic scattering problems, Green's theorem specifies that the total acoustic pressure p (p_{inc} + p_s, with p_{inc} defined as the incident pressure on a surface S due to sources in the region of interest and p_s defined as the complex scattered pressure exterior to S) and the normal velocity v satisfy the Helmholtz integral formulas:

$$\int_S \left\{ p(\sigma) \frac{\partial}{\partial n_\sigma} G(x,\sigma) + i\omega\rho v(\sigma) G(x,\sigma) \right\} ds(\sigma) = \tag{1}$$

$$\begin{cases} p(x) - p_{inc}(x) \;\; ; \;\; x \text{ exterior to } S \\ \frac{1}{2} p(x) - p_{inc}(x) \; ; \; x \text{ on } S \\ - p_{inc}(x) \; ; \; x \text{ interior to } S \end{cases}$$

with:

$$G(x,x') = \frac{1}{4\pi} \frac{e^{-ik|x-x'|}}{|x-x'|} \qquad \text{and} \qquad k = \frac{\omega}{c} \;\; ; \tag{2}$$

where:

$G(x,x')$ = three dimensional free-space Green's function and

k : acoustic wave number

ω: radian frequency

c: sound speed in the fluid

$\frac{\partial}{\partial n}$: signifies differentiation in the outward normal direction

and a harmonic time dependence of $e^{i\omega t}$ is assumed.

If ζ is some point located on the surface, equation (1) from above leads to:

$$\frac{1}{2}p(\zeta) - \int_s p(\sigma)\frac{\partial}{\partial n_\sigma}G(\zeta,\sigma)ds(\sigma) = p_{inc}(\zeta) + i\omega\rho\int_s v(\sigma)G(\zeta,\sigma)ds(\sigma) \quad (3)$$

C. CHIEF – NUMERICAL APPROACH

Since the goal in CHIEF is to solve this problem numerically, the surface S is broken up into an N_S number of non-overlapping "patches". The pressure and velocity at each of these patches is approximated by average values, designated as P_n and V_n, respectively.

The discretization of equation (3) leads to:

$$\frac{1}{2}P_m - \sum_{n=1}^{N_s} P_n \int_{S_n} \frac{\partial}{\partial n_\sigma}G(\zeta_m,\sigma)ds(\sigma) = p_{inc}(\zeta_m) + i\omega\rho\sum_{n=1}^{N_s} V_n \int_{S_n} G(\zeta_m,\sigma)ds(\sigma) \quad (4)$$

$$m = 1,\ldots,N_s,$$

The matrices **A** and **B** are subsequently defined as follows:

$$A_{mn} = \frac{1}{2}\delta_{mn} - \int_{S_n} \frac{\partial}{\partial n_\sigma} G(\zeta_m, \sigma) ds(\sigma)$$
$$B_{mn} = i\omega\rho \int_{S_n} G(\zeta_m, \sigma) ds(\sigma), \qquad (5)$$

If the variables **p** , **v**, and $\mathbf{p_{inc}}$ are defined as vectors of surface pressures, velocities and incident surface pressures, then equation (4) can be more simply stated as:

$$\mathbf{A\ p = B\ v + p_{inc}} \qquad (6)$$

Note: further refinement of the **A** and **B** matrix definitions in equation (5) above are required to handle specific wave numbers and surface pressure conditions. The matrix **A** is otherwise not suited to achieving an over determined system solvable via least squares solution technique. This further refinement is not addressed here.

As mentioned in the early part of this section, the Helmholtz Integral relations can also be used to deduce pressures at defined locations in space. The pressure at any field point x (near field or far field) can be expressed in terms of the surface pressure and normal velocity per the following relationship:

$$p_s(x) = p(x) - p_{inc}(x) \qquad (7)$$

$$= \int_S \left\{ p(\sigma) \frac{\partial}{\partial n_\sigma} G(x, \sigma) + i\omega\rho v(\sigma) G(x, \sigma) \right\} ds(\sigma)$$

In like fashion as the formulation of the CHIEF **A** and **B** matrices, it desired to solve equation (7) numerically by breaking the surface S into a mosaic of non-overlapping patches. This leads to the expression identified in equation 8 that follows.

$$p_s(x) = p(x) - p_{inc}(x) \tag{8}$$

$$= \sum_{n=1}^{N_S} P_n \int_{S_n} \frac{\partial}{\partial n_\sigma} G(x,\sigma) ds(\sigma) + i\omega p \sum_{n=1}^{N_S} V_n \int_{S_n} G(x,\sigma) ds(\sigma)$$

Using the condition where $p_{inc}(x)$ is equal to zero (indicating that there is no incident pressure on the surface), the discrete notation of equation 8 may be further simplified to:

$$p(x) = \mathbf{p\ delG + \ v\ G} \tag{9}$$

where each of the constituents (from left to right) on the rhs of equation 9 are vectors of the pressure at a known surface location, the integration of the normal derivative of the Greens function for the given set-up, the normal velocity at a known surface location and the integration of the Greens function for the given set-up, respectively. Please note that x designates the three dimensional location in space of a defined field point and σ (equation 8) designates the three-dimensional location in space of the surface patches.

D. SUMMARY

CHIEF utilizes the Helmholtz integral relations of surface pressures and normal velocities along with a defined surface incident pressure condition. Computations within CHIEF proceed numerically via the re-structuring of the continuous surface into a mosaic of individual non-overlapping surface patches. This re-structuring allows the formulation of the generalized equation:

$$\mathbf{A\ p = B\ v + p_{inc}} \tag{10}$$

with **p** , **v**, and $\mathbf{p}_{inc}$ defined as vectors of surface pressures, normal velocities and incident surface pressures, respectively. **A** is the matrix formed from surface integrals of the normal derivative of pressure over the surface patches and **B** is the matrix formed from surface integrals of the pressure over the surface patches. Both the **A** and **B** matrices can be formed from known acoustic constants and a defined geometry [2] .

CHIEF also yields a mechanism to calculate pressure at a defined field point via the use of known surface pressures and normal velocities and integrated three-dimensional Greens function derived relationships (between surface locations and the defined field point x). This leads to the following relationship:

$$p(x) = \mathbf{p\ delG} + \mathbf{v\ G} \qquad (11)$$

with $p(x)$ defined as the pressure at a designated field point x, **p** and **v** defined as vectors of surface pressures and normal velocities, respectively, and **delG** and **G** defined as the integrated Greens functions relationships between field point x and the surface patch locations.

Equations 10 and 11 in the above summary form most of the basis that will be utilized in setting up the near field-far field extrapolation problem's defining equations. The methodology will require that Equation 10 be equated to zero (indicating that no external sources are present) and equation (11) will be used to relate to pressures at defined near field points.

Due to the numerical approach that CHIEF utilizes, it is important that the number of patches used in the mosaic to cover the surface, S, be properly sized. A single complex value of pressure and velocity per patch location is a required condition in CHIEF; these values of pressure and velocity over the entire patch are represented by values computed for the patch center.

IV. STEPS OF THE METHODOLOGY

A. INTRODUCTION

This section outlines and details the steps of the methodology required to take near field array element data through the processing steps required to achieve an extrapolated far field response. The steps provide the procedural details as well as criteria for parameter selection. Practical considerations of limitations of the processes are also a necessary part of these steps since the effort entails the implementation of the technique on available computing platforms. This section is supplemented by the "Explanation of MATLAB Coding" section that appears in Appendix C.

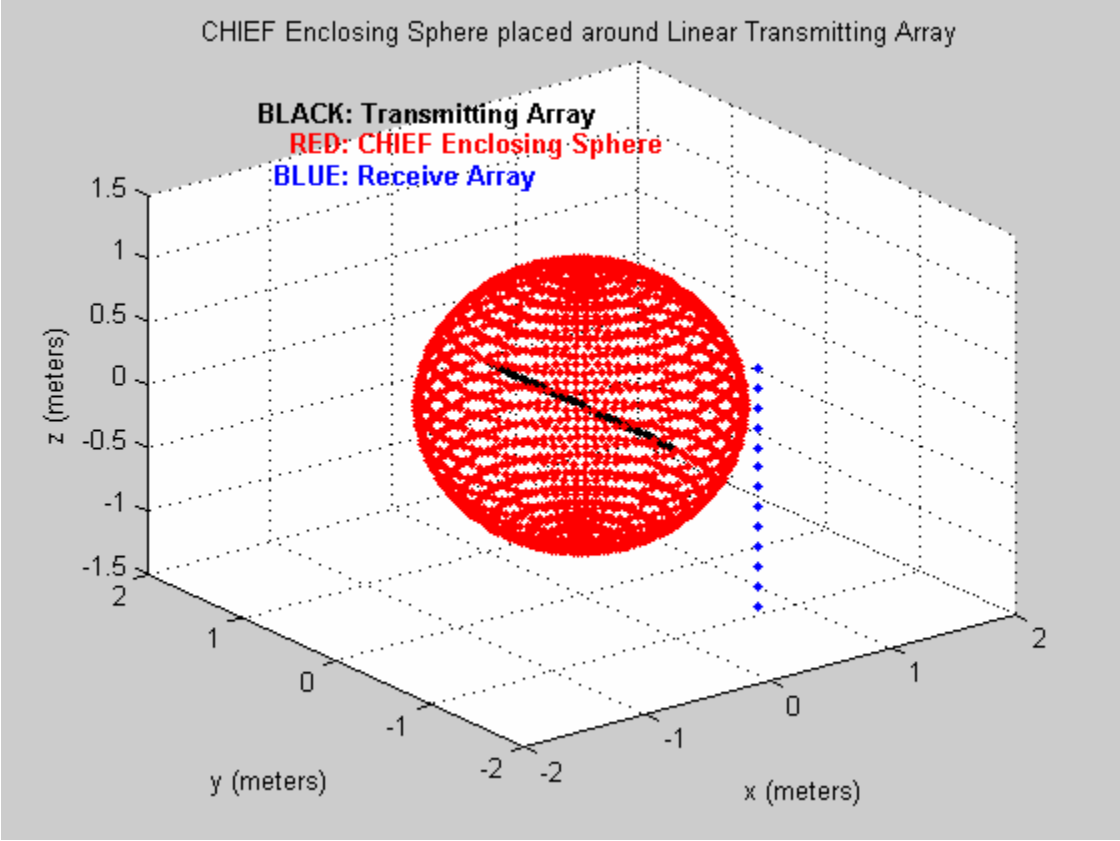


Figure 3 – Typical Methodology Set-up showing device under test, receive array and CHIEF Enclosing Sphere

The figure above shows a typical set-up that forms the basis of the methodology. The transmitting array and the receive array both represent actual physical devices, while the CHIEF enclosing sphere represents an artificial surface in space. The methodology seeks to define the pressure and velocity characteristics of this spherical surface for eventual extrapolation to far field pressure.

While some of the early steps are not execution order specific, it is practical to treat all of these steps as sequential in the order that they are presented.

B. DEFINE CHIEF ENCLOSING SPHERE

This step of the process takes into account the physical size of the acoustic device, and its frequencies of operation. Since the enclosing sphere (formulated using "CHIEF for MATLAB" protocols [2]) will be formed from multiple repeated copies of a single "slice" of a sphere (imagine a segment from a section of a tangerine), an estimate of the number of comprising slices required, ie: its symmetry order (sph_sym), is performed in this step. The number of divisions within this single "slice" of the sphere (designated as patches) are also formed in this step. The slice will, therefore, be represented by a sph_numv x 1 matrix with sph_numv representing the number of equally spaced patches in the direction perpendicular to the symmetry order . (Limitations are imposed on both of these quantities based upon computational capabilities and efficiencies of both the MATLAB coding and the standard COTS computer hardware used for this effort.)

C. DEFINE RECEIVE ARRAY AND PROXIMITY TO ACOUSTIC DEVICE

This step defines the number of receive array elements (numh) and the array's distance from the acoustic device under test. It is assumed that the elements in this array are in a vertical line and are equally spaced. The technique does not necessarily require this for any particular technical reason, rather this is based more upon the expected configuration that will be suited for an experimental set-up. Maximum spacing of the elements in this array is expected to be dependent upon the required frequencies of

operation. The maximum number of receive elements for this technique will be established more by logistical considerations rather than for technical reasons (ie: there will be some practical limit to the number of elements that can be made available for the receive array based upon available sensors in inventory and the number of channels available on the data acquisition system). The distance of the receive array from the acoustic device will be in the near field of that device. Part of the evaluation of this technique will be to examine final extrapolated data to assist in determining the robustness of the technique with decreasing separation distance from the acoustic source and varying receive array element spacings and numbers. This technique will require that the receive array be placed outside the boundaries imposed by the enclosing CHIEF sphere.

D. RECEIVE ARRAY MEASUREMENTS/SIMULATION

This step involves the acquisition (or simulation) of complex receive response at each of the receive array elements. This step requires that data be collected for each of the receive array elements at acoustic device rotation angle intervals corresponding to (360 degrees /symmetry order of the enclosing sphere). The sampling (or later interpolation) of this data at the prescribed angles allows a direct relationship between the data collected and the defined "slices" of the enclosing sphere (established by the selected symmetry order).

E. COMPUTE ORIGINAL A AND B MATRICES VIA CHIEF FOR MATLAB

This step involves the computation of the original **A** and **B** matrices using "CHIEF for MATLAB" tools [2] . The **A** and **B** matrices are formed from the surface integrals in CHIEF. The following is taken from documentation provided for "CHIEF for MATLAB" by S.E. Forsythe (NUWCDIVNPT): "The standard problem statement for any CHIEF problem is: $\mathbf{Ap} = \mathbf{Bv} + \mathbf{p}_{inc}$, where **A** is the matrix formed from surface integrals of the normal derivative of pressure over the surface patches, **B** is the matrix formed from surface integrals of the pressure over the surface patches, $\mathbf{p}_{inc}$ is the vector

of incident pressures on the surface patches due to sources in the region of interest, **p** is the (to be solved for) vector of total pressures on the surface, and **v** is the (to be solved for) vector of velocities on the surface."

CHIEF for MATLAB has the capability to provide the **A** and **B** matrices and cease further computations (to solve for resulting pressures and velocities). This is desired for our far field extrapolation technique since the **A** and **B** matrices will be augmented with other factors in the next step. The primary inputs to create the **A** and **B** matrices depend upon the acoustic parameters of sound speed, density of the test medium and frequency of interest and upon the geometric configuration of the surface of interest. In our case, the surface of interest is the enclosing sphere from the first step. Thus, given a defined enclosing sphere, the matrices **A** and **B** can be formed.

F. COMPUTE AUGMENTED MATRICES A_A , B_A AND M_A

This next step begins the departure from the usual manner in which the CHIEF program is utilized. That is, typical CHIEF applications require the definition of a transducer geometry followed by the characterization of a pressure, velocity or impedance across all defined patches. Specification of one of these parameters (pressure, velocity or impedance) is required to make a reduced system (from the original defining equations of $\mathbf{Ap} = \mathbf{Bv} + \mathbf{p_{inc}}$ that is uniquely solvable.

In this thesis methodology, the only completely defined geometries are represented by the CHIEF enclosing sphere (an artificial surface) and the physical location of the receive array elements; the geometry of the device under test is not required to be known exactly (with the exception that its largest dimension must be known). This flexibility to work with a minimal set of known transducer geometries was a desired feature of the methodology.

The other significant difference in this CHIEF usage is that the pressure and velocity terms at each patch location on the defined CHIEF enclosing sphere are both unknown values that must be solved for using additional information constraints.

A least squares solution [7] is the ultimate goal of the methodology and relies upon the use of field values of pressure as the starting point. The relation between the CHIEF **A** and **B** matrices and pressure and velocity terms form the fundamental relationship of $0 = -\mathbf{A}\ \mathbf{p} + \mathbf{B}\ \mathbf{v}$ and is examined first below. Since this system by itself is underdetermined, the system is augmented by other relationships to achieve overdetermined status. The augmentation relies upon the CHIEF **A** and **B** matrices and their relationship to field pressures; reliance is also made upon expected adjacency conditions of pressure and velocity between patches.

The **A** and **B** matrices formed in the last step provide the starting point to develop the following relationship:

$$\mathbf{M_a} = -\mathbf{A_a}\,\mathbf{p} + \mathbf{B_a}\,\mathbf{v}$$

where:

$\mathbf{A_a}$ = Augmented **A** Matrix

$\mathbf{B_a}$ = Augmented **B** Matrix

p = pressures at all "patches" on the enclosing sphere (a vector)

v = velocities at all "patches" on the enclosing sphere (a vector)

$\mathbf{M_a}$ = Augmented Field Points Matrix (complex responses at defined points in the test medium)

The augmentation of the matrices in this step puts the measured data and defining physical equations in a form that will be suited for a later least squares solution determination. The augmentation of the **A** and **B** matrices is examined first and is described as follows:

1) 0 = **-A p** + **B v**

This relationship states that the computed **A** and **B** matrices (from the geometrical and acoustic considerations of the previous step) when multiplied by their respective pressures and velocities result in zero net pressure on the surface when there is no source in the region of interest as required by the Helmholtz theorem.

2) pressure at a field point = **delG p** + **G v**

This defining equation relates the Greens function [8] and its gradient along with velocity and pressure values at each physical location on the enclosing sphere to complex response values measured at (near) field test points. The Green's function and its gradient are determined via geometrical considerations (ie: the physical location of the field points with respect to the enclosing sphere). The pressures at field points are as measured (or simulated) by the receive array. As in the first step of this augmentation, this defining equation utilizes the (to be solved for) pressures, **p**, and velocities, **v**, along with factors that can be determined (**G** and **delG**).

3) pressure/velocity equivalence at adjacent patches

This relationship implies that due to the relatively small size of the patches comprising the surface of the enclosing sphere, it is reasonable to assume that adjacent patches do not differ significantly in pressure or velocity from one another. They can, in fact, be considered to be approximately equal. This allows a physical attribute to be associated with the (to be solved for) pressures and velocities. Enforcing this condition with a small weight in the least squares solution prevents the extreme case where non-realizable pressure and velocity swings are required for mathematical realization [7]. Thus, this step essentially states that $p_i = p_{i+1}$ and $v_i = v_{i+1}$, where i represents some patch location and i+1 represents the patch next to it in the vertical direction.

The resulting augmentation of the **A** and **B** matrices, therefore, yields the following constituents:

$\mathbf{A_a}$ **Matrix**	$\mathbf{B_a}$ **Matrix**
- A p	**B v**
delG p	**G v**
$p_i = p_{i+1}$	$v_i = v_{i+1}$

Note that pressure terms are related to the $\mathbf{A_a}$ matrix while velocity terms are related to the $\mathbf{B_a}$ matrix. (The details of the actual construction protocols of these matrices are examined later.)

The construction of the augmented field point array, $\mathbf{M_a}$, is now examined. The $\mathbf{M_a}$ matrix represents the expected outputs or measured values corresponding to the constituents of the augmented $\mathbf{A_a}$ and $\mathbf{B_a}$ matrices. For the three defined parameters discussed above, $\mathbf{M_a}$ takes on the following values:

1) $\mathbf{M_a} = 0$

for the (-Ap+Bv) terms

2) $\mathbf{M_a}$ = complex response measured (simulated) at each element of the receive Array for the (**delG p** + **G v**) terms

3) $\mathbf{M_a} = 0$

For the adjacent equivalence terms of pressure and velocity represented by $(p_i - p_{i+1})$ and $(v_i - v_{i+1})$

Since a least squares solution technique will be utilized in the next step, an advantage in optimizing the solution is gained by scaling the matrix constituents. These re-scaled matrix components still accurately represent the conditions and equalities of the set-up. They do, however, de-emphasize the wide variation in magnitudes of the various quantities and allow the solution criteria to proceed based upon more equal weightings. This re-scaling is also appropriate to identify as putting the defining equations in dimensionless form.

For example, the Green's Function integral, **G**, for a particular set-up returns an absolute maximum value of 573 while the gradient returns an absolute maximum value of 3.82e-4. Adjusting the Green's Function integral elements by first dividing all **G** elements by rhoc (1500000) and then dividing all matrix elements of (the new) **G** and **delG** by 3.82e-4 gives matrix elements on both **G** and **delG** that have maximum values of approximately one.

While the above protocol establishes more or less equal weightings, the methodology does allow the user the opportunity to place specific emphasis on the contributions of the theoretically defined Helmholtz integral relations or of the empirically acquired near field measurement data. This is embodied in the alpha and beta terms defined below.

For this effort, the comprising matrices are scaled by the following factors:

Matrix	Scaling Factor
A	alpha
B	alpha/rhoc
delG	beta
G	beta/rhoc
M	beta

with:

alpha = alpha0 / (maximum absolute value in **A**)

beta = beta0 / (maximum absolute value in **delG**)

and:

alpha0 and beta0 are as defined by the user to emphasize the theoretical terms (Helmholtz integrals relations) or emphasize the empirical data (near field measurements), respectively, in the final solution.

This completes the augmentation of the defining matrices. It is significant to note at this point the expected size of the augmented matrix constituents. This information is tallied below:

Matrix	Size (rows x columns)
A	numv x (sph_numv * sph_sym)
B	" "
delG	numh x (sph_numv * sph_sym)
G	" "
Pressure equivalence	(sph_numv-1) x (sph_numv * sph_sym)
Velocity equivalence	" "
$\mathbf{M_a}$	(sph_numv + numh + 2*sph_numv -2) x 1

The matrix sizes presented here do not constitute the full matrix, rather they are representative of one "slice" of the full circulant matrix [2] . The full matrix size is larger by a factor of sph_sym for each of the above constituents.

The full **Aa** and **Ba** matrices are, therefore, of size:

(sph_sym * (numh + 3*sph_numv -2)) x (sph_numv * sph_sym)

while the full **Ma** matrix is of size:

(sph_sym * (numh + 3*sph_numv -2)) x 1

G. SOLVE FOR PRESSURES AND VELOCITIES VIA LEAST SQUARES

This step uses the least squares solution [7] technique to compute the pressure and velocity values at each of the enclosing sphere patches. The output for this step consists of (numv * sph_sym) pressure values and the same number of velocities.

The least squares technique allows an overdetermined system ("m" equations in "n" unknowns, with m>n) to be evaluated in an optimized manner. This optimized manner results in values of x that minimize the quantity $|\mathbf{Ax} - \mathbf{b}|^2$ for the originating equation of $\mathbf{Ax} = \mathbf{b}$.

The vector **x** that minimizes $|\mathbf{Ax} - \mathbf{b}|^2$ is the solution to the normal equations:

$$\mathbf{A}^T \mathbf{Ax} = \mathbf{A}^T \mathbf{b}$$

This vector $\mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}$ is the least squares solution to $\mathbf{Ax} = \mathbf{b}$.

The implementation of the least squares technique for this thesis effort substitutes a matrix that will be designated as the **AB_Matrix** for **A** and the full $\mathbf{M_a}$ matrix for **b**. The **AB_matrix** consists of the full matrices $\mathbf{A_a}$ and $\mathbf{B_a}$ that have been interleaved as follows:

$Aa_1\ Ba_1\ Aa_2\ Ba_2\ Aa_3\ Ba_3 \ldots\ Aa_n\ Ba_n$; with n = (sph_numv * sph_sym)

The **AB_Matrix** is actually a circulant matrix of size:

(sph_sym * (numh + 3*sph_numv -2)) x (2 * sph_numv * sph_sym)

The justification for the interleaving of the $\mathbf{A_a}$ and $\mathbf{B_a}$ components will be examined in detail later.

The least squares solution, **x**, given both **AB_Matrix** and $\mathbf{M_a}$, is a vector of alternating pressures and velocities (of total length: 2 * sph_numv * sph_sym). These values correspond to the pressures and velocities at each patch on the enclosing sphere.

H. COMPUTE $M_{A\text{-}NEW}$ MATRIX

This step takes the pressures and velocities computed in the last step and substitutes them into the following equation to compute the new augmented field point matrix, $\mathbf{M_{a\text{-}new}}$.

$$\mathbf{M_{a\text{-}new}} = -\mathbf{A_a}\,\mathbf{p} + \mathbf{B_a}\,\mathbf{v}$$

The $\mathbf{A_a}$ and $\mathbf{B_a}$ matrices are the same as that defined previously in step #5. The **p** and the **v** represent the pressure and velocities computed via the last step.

This newly computed matrix, $\mathbf{M_{a\text{-}new}}$, can then be used to evaluate the validity of the enclosing sphere pressures and velocities, **p** and **v**, and whether the solution is likely to be properly optimized. This is accomplished by comparing $\mathbf{M_{a\text{-}new}}$ quantitatively to the original $\mathbf{M_a}$ matrix.

Since $\mathbf{M_{a\text{-}new}}$ utilized a data set derived from a least squares solution (the **p** and **v** values), it is not expected that it will match up exactly one for one with the originating $\mathbf{M_a}$ matrix. There are important considerations to be made, however.

Before listing this criteria, a short summary of the constituents of the $\mathbf{M_a}$ matrix is first reiterated. The $\mathbf{M_a}$ matrix is divided up into four sections.

1) The first section has values corresponding to the (**-Ap** + **Bv**) calculations. Ideally these values should be zero.

2) The second section contains the complex response data at each element of the receive array (these are near field data points). These are the terms of (**delG p** + **G v**).

3) The third section contains the pressure adjacency terms (p_i - p_{i+1}). These values should be close to zero.

4) The fourth section contains the velocity adjacency terms (v_i - v_{i+1}). These values should be close to zero.

The following are the primary criteria that impact the successful use of this technique for far field extrapolation.

1) The complex response data (section two) of the two matrices, $\mathbf{M_{a\text{-}new}}$ and $\mathbf{M_a}$, should agree with one another as closely as possible.

2) The (**-Ap** + **Bv**) term (section one) of the $\mathbf{M_{a\text{-}new}}$ matrix should be minimized as much as possible since this represents a known physical parameter.

3) The pressure adjacency and velocity adjacency terms (sections three and four) are expected to be small terms, but this condition does not require a heavy enforcement. Some differences in pressure and velocity values on adjacent patches are expected. This criteria was originally added to lessen the chance of skewing the pressure and velocity values to physically unrealizable states.

4) Since "zeros" are not likely to be obtained for sections 1,3 or 4, it is important to consider the difference (in dB) between the magnitudes in these sections and the magnitudes of the complex response data at each element of the receive array (section 2). This difference should be maximized as much as possible.

The data presented below shows the two matrices, $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$, for a given set-up. This data presented has been optimized to a reasonable level (as examined later) and would serve to qualify the solved-for pressure and velocities for the last step of this methodology, the step when the far field computations are actually made.

The four distinct sections are most easily observed in the $\mathbf{M}_{\mathbf{a\text{-}new}}$ figure. The sections 1 thru 4, as defined above, appear in succession from the top of the figure to the bottom. The figure shows magnitudes expressed in dB; the colorbars to the right of the figures designate the correspondence between color and magnitude. The reddish band of colors correspond to the complex response data (section 2) and, as expected, are the highest magnitudes in the matrix.

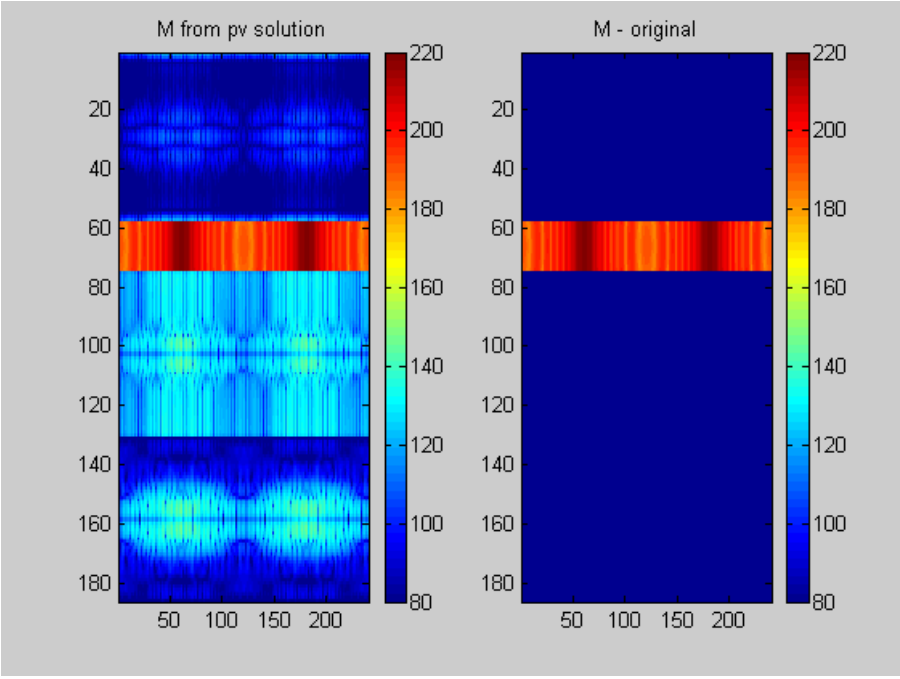


Figure 4 – Original and "pv solution" Augmented Field Point Matrices ($\mathbf{M}_{\mathbf{a}}$ and $\mathbf{M}_{\mathbf{a\text{-}new}}$)

I. COMPUTE FIELD POINT RESPONSES

This last step uses the pressure and velocity data obtained and evaluated from the previous two steps to compute complex response at defined field points. This computation proceeds via the use of the Greens function integral and its gradient as has been accomplished in earlier steps when defining the augmented matrices. This familiar relationship is:

$$p = (\mathbf{delG}\ \mathbf{p} + \mathbf{G}\ \mathbf{v})\ ^{[2]}$$

As before, the **p** and **v** values represent the pressures and velocities at each patch location on the enclosing sphere. The **G** and **delG** terms represent the Greens function integral and its gradient obtained at specified field points with respect to the enclosing sphere. The selection of field points will typically be made to encircle the acoustic device on a designated plane allowing what are traditionally called XY, XZ, or YZ directivities.

It is significant to note here that there is no distinction in this step whether defined field points are in the near field or in the far field. This technique takes a pressure/velocity defined enclosing sphere and combines its elemental traits via known physical relationships (Greens function integral and its gradient) to achieve pressure values at defined locations. Normalizing directivity data obtained via this technique for an increasing range will show the expected convergence to a single far field pattern of defined character.

V. SOLUTION SET USING METHODOLOGY

This section takes the theory and methodology detailed in the previous sections to solve a real world problem.

A. DEFINITION OF SET-UP

In this case (and as supported by the referenced figures, Table 1 and the MATLAB coding provided), a horizontal transmitting line array of length 1.885 meters is operated at a frequency of approximately 4775 Hz. The expected near field/far field boundary for this device occurs at a distance of approximately 11.32 meters. This distance exceeds the physical size of the Acoustic Pressure Tank Facility (APTF), 11.1 meters in length, where this device's performance under temperature and pressure conditions are desired to be characterized. It is an ideal candidate to investigate the methodology detailed within this thesis.

Figures 8 and 9 show the expected theoretical near field and far field responses, respectively, of the defined device. Figure 8 shows near field performance at a range of approximately 5.341 meters. This test distance is within the available range of separation distances that can be achieved within the APTF. The theoretical far field response was computed at a test distance of about 1131 meters. This places device performance characterization well into far field conditions. (Please note that the designation of element #6 on the near field data , Figure 8, corresponds to response in the XY plane. As will be described later, element #6 is the mid-point element of the 11 element vertical line array used in this effort.)

A vertical near field array of approximate length 1.571 meters was used for this set-up. This near field array is made up of 11 equally spaced elements and was placed at a distance of 5.341 meters from the origin. (The horizontal transmitting array will be rotated about this same origin in the XY plane for this set-up. This origin location corresponds to the mid-point of the transmitting array.) The lengths specified for both

the near field array and for the transmitting array fit within the confines of the 3.81 diameter APTF test vessel.

The selection of 11 elements for the near field array was arrived a bit by experimentation and by a concerted effort to reduce the numbers of acquisition channels that ultimately would be required for this implementation (ie – 16 channel acquisition cards are fairly easy to procure as off-the-shelf components and would be suited to a future hardware implementation for this technique).

Figure 7 shows a section of the enclosing CHIEF sphere that was used. The enclosing sphere was configured to have a radius of about 1.384 meters. The sphere section displayed in Figure 7 is made up of 84 individual patches. A symmetry order of 180 was selected; this yields a total of 15,120 patches on the enclosing sphere surface. It is a part of this methodology to solve for a single pressure and a single velocity at each one of these patches. Thus, 30,240 "unknowns" must be solved for in this case. (As will be discussed later, the number of individual patches required on the enclosing surface is the primary limiting factor for this methodology. High frequency devices with large apertures will likely require an enclosing sphere of relatively large size yielding an extreme number of patches. This likely prohibits solution via this implementation on standard personal computing hardware and available COTS software.)

B. INITIAL TEST RESULTS

Figures 10 through 14, inclusive, show outputs obtained via the second step of the MATLAB coding. Figure 10 shows a comparison in absolute magnitude of the original data set (obtained via simulation in this example but would correspond to actual measured data in a real world implementation) and the hydrophone near field response obtained via the technique. These latter hydrophone responses were obtained using the solved for pressure and velocities from the technique. Figure 10 actually shows the difference between these two data sets. All differences are less than |0.015| dB showing an excellent match.

The remaining figures, 11 thru 14, allow the operator to assess the other conditions of the methodology. Specifically, the values computed for the Helmholtz Integral relations and those corresponding to the pressure and velocity adjacency conditions need to be assessed as well to judge acceptance of the solved for pressures and velocities. In an ideal solution, these other conditions would be equal to zero. Since the least squares solution technique was utilized, a best fit to the defining equations was determined based upon near field array element data and assumed zero conditions for the Helmholtz Integral relations and pressure/velocity adjacency conditions. Is it not a true requirement to have these latter two conditions exactly equal to zero. It is, however, desirable to have these other conditions much smaller than the computed hydrophone responses. It is for this reason that dB levels are reported in Figures 11 through 14.

Figure 11 shows a plot of the minimum hydrophone responses, maximum Zero0 and maximum Zero1 conditions on a per angle basis. The nomenclature Zero0 and Zero1 correspond to the "expected zeros" for the Helmholtz Integral relations and pressure/velocity adjacency conditions, respectively. Since the need here is to assess the solved for pressures and velocities, it is desirable to look at worst case conditions in these data sets. That is, the selection of minimum hydrophone response and maximum Zero0 and Zero1 responses provides the smallest difference between the two data set conditions. (Reference Figure 3 – Original and "pv solution" Augmented Field Point Matrices ($\mathbf{M_a}$ and $\mathbf{M_{a\text{-}new}}$) for visualization of the Zero0 and Zero1 conditions. Zero0 values are shown above the reddish band, the complex response data, while Zero1 values are shown below.)

Figures 12 and 13 augment Figure 11 by providing the actual computed differences described above. In this data set, a minimum difference on the order of about 44 dB and 54 dB are observed for the Zero0 and Zero1 conditions. These magnitudes, from experimentation, are likely to indicate that the solved for pressures and velocities are acceptable to use for far field extrapolation. The adjustment of the values for the alpha0 and beta0 terms will be looked at later to try and better this solution.

Figure 14 is perhaps the most useful of all of the figures in the Figure 11 through 14 set. Figure 14 provides a "snapshot" of all of the parameters discussed above (Helmholtz Integral Relations, Near Field Hydrophone Response, Pressure Adjacency and Velocity Adjacency conditions). It provides a means to directly compare these parameters with their originating values. It also allows the operator to make direct comparison amongst the $M_{a\text{-}new}$ constituents (which are the parameters mentioned above). The color in this figure is representative of magnitude as shown in the colorbar scale provided. Thus, it is possible to roughly deduce at a glance, all of the differences as noted in Figure 11 through 13.

Figures 15 through 18 used the solved for pressures and velocities obtained for alpha0=1 and beta0=1 (with the performance characteristics as identified in the previous figures) to solve for pressure at defined field points. These defined points correspond to the original near field points and points in the far field (which is the motivation for this technique). Figures 15 and 16 show that the solved for pressures and velocities yield near field pressures that have an extremely high degree of agreement with the original near field data set. Absolute differences of less than .008 dB were observed.

Figures 17 and 18 show far field response as originally simulated and as extrapolated by this technique. The majority of the differences in these two data sets (Figure 18) are on the order of a few tenths of a dB with just a few outliers (these are addressed later). Thus, the technique has done an excellent job of matching expected far field performance with that obtained via the use of solved for pressures and velocities on an enclosing sphere. The technique is therefore viable.

As mentioned above, the majority of the differences in the two data sets (Figure 18) are on the order of a few tenths of a dB. There are, however, some difference values as high as about 4.3 dB. Closer examination of the locations of these large differences shows a correspondence with a null in the directional response. Thus, the technique did an excellent job of producing a far field response but missed representing to a high

accuracy the depth of the nulls in these far field responses. This will be looked at again once new values of alpha0 and beta0 are utilized.

The values for alpha0 and beta0 are now changed to evaluate their effect on the solved for pressures and velocities and eventually their effect on extrapolated far field performance.

C. SENSITIVITY OF SOLUTION TO VARIATION IN BETA0

Figures 19 through 23 used values of alpha0=1 and beta0=10. (Recall that the beta0 term emphasizes near field element data.) Figure 23 shows some improvement in the difference in level between the hydrophone response measured and its other zero0 and zero1 constituents. These minimum differences are now about 74 and 64 dB, respectively. Pressure computations at the various field points were not computed.

Figures 24 through 28 used values of alpha0=1 and beta0=100. An improvement in difference levels was noted (97 dB and 84 dB) , however, pressure computations were not undertaken.

Figures 29 through 33 used values of alpha0=1 and beta0=500. Differences of 110 dB and 98 dB between hydrophone response and Zero0 and Zero1 conditions, respectively, were noted. Figure 33 shows this same level of difference by inspection.

Despite changes in magnitude between these various $\mathbf{M}_{\mathbf{a\text{-}new}}$ figures, an examination of Figure 33 and Figure 14 shows how the beta0 term is forcing a larger difference between the hydrophone response data and the Helmholtz Integral relations and pressure/velocity adjacency conditions. Each of these $\mathbf{M}_{\mathbf{a\text{-}new}}$ figures has a dynamic range of 150 dB. Figure 33 shows how the hydrophone response data is being emphasized while the other parameters are being suppressed.

Computation of pressure at the various field points was undertaken for the alpha0=1 and beta0=500 conditions. Figures 34 through 37 show the resulting pressure

responses and simulated/technique derived differences for these values. As before, the near field data continues to show solid agreement between simulated and technique derived values (Figures 34 and 35). Figures 36 and 37 show far field performance traits. The noteworthy part of this data set is that the differences between the two far field data sets are now less than 1 dB (down from 4.3 dB in Figure 18). Thus, this beta0 value change has resulted in pressures and velocities that provide a better far field fit. This is especially seen now in the better agreement afforded at the nulls of the far field response.

A final beta0 change to beta0 = 1000 was undertaken. Figures 38 through 43 show data obtained with alpha0=1 and beta0=1000. As can be seen on the far field difference reported in Figure 43, there was little change in far field response provided by this.

While far field agreement in this case could have essentially stopped with alpha0=1 and beta0=1, the other values of beta0 were examined to evaluate the technique. Increases in beta0 did lead to a better fit in the nulls of the far field, but this is likely of limited practical value.

D. SENSITIVITY OF SOLUTION TO VARIATION IN ALPHA0

In likewise fashion, to further evaluate the technique, changes in the value of alpha0 (emphasizing the theoretical portion of the technique) will now be undertaken.

Figures 44 through 48 show data corresponding to alpha0=10 and beta0=1.

Figures 49 through 53 show data corresponding to alpha0=100 and beta0=1.

Figures 54 through 58 show data corresponding to alpha0=100 and beta0=1.

An examination of the $\mathbf{M}_{\mathbf{a\text{-}new}}$ figures for each of these conditions shows that there is little to no change in level differences between hydrophone response and

pressure/velocity adjacency conditions. The Helmholtz Integral relations are showing a suppression as alpha0 increases.

Figures 59 through 62 show pressure responses computed for the near field and far field cases. An examination of the far field difference figure (Figure 62) shows that very little was gained by increasing the alpha0 term. The differences in the nulls that were noted for the alpha0=1 and beta0=1 far field computations still exist and are at the same approximate level.

E. SUMMARY OF SENSITIVITY OF SOLUTION TO ALPHA0 AND BETA0

Table 1 summarizes the conditions examined in the previous examples. The following definition of terms applies to the data presented.

alpha0:

This is the weighting term to emphasize the theoretical Helmholtz Integral Relations.

beta0:

This is the weighting term to emphasize empirical data

MAX |Near Field Hydro Diff|:

This value represents the maximum difference between the measured hydrophone responses of all near field receive array elements and the values computed by the second step of the methodology. A low value represents good agreement and, therefore, a possible candidate for far field extrapolation (using the pressure and velocity values so obtained).

MIN min_hydro_minus_max_zero0:

This value represents the minimum difference between the minimum near field hydrophone data and the maximum zero0 data (theoretical Helmholtz relations) obtained at each angular position. A large magnitude here also signifies that the conditions are favorable for far field extrapolation.

MAX min_hydro_minus_max_zero0:

This value represents the maximum difference between the minimum near field hydrophone data and the maximum zero0 data (theoretical Helmholtz relations) obtained at each angular position. This is largely for informational purposes only.

MIN min_hydro_minus_max_zero1:

This value represents the minimum difference between the minimum near field hydrophone data and the maximum zero1 data (pressure and velocity equivalency for adjacent patches) obtained at each angular position. A large magnitude here also signifies that the conditions are favorable for far field extrapolation.

MAX min_hydro_minus_max_zero1:

This value represents the maximum difference between the minimum near field hydrophone data and the maximum zero1 data (theoretical Helmholtz relations) obtained at each angular position. This is also largely for informational purposes only.

MAX |Far Field Difference|:

This value represents the maximum difference between the methodology-computed far field response and a measured (or in this case, simulated) far field response obtained at each angular position. A low value indicates that the methodology produced

a reasonable approximation to actual far field performance. A high value may indicate an overall poor match or perhaps just a poor representation at certain angles (most likely in the nulls). It is advantageous to look at the overall pattern obtained to further define the severity of a high value. (Please note, however, that all constituents to obtain this data value would not typically be available in practice (ie – the purpose of this thesis was to achieve a far field formulation when only near field test conditions were available). These results were available in this thesis effort and therefore have been examined.)

The initial equal weighting of the alpha0 and beta0 terms (by setting both equal to 1) for this example gave excellent results with little gained by any other adjustments (other than a bit better definition of the response in far field nulls). This is not necessarily likely to be accurate in all cases. Where there is some uncertainty in the measured data acquired, a better fit may be obtained by emphasizing the theoretical portion of the technique. This will be examined in the next section when error in range and positioning of the near field array are examined.

alpha0	beta0	MAX |Near Field Hydro Diff| (dB)	MIN min_hydro_minus_max_zero0 (dB)	MAX min_hydro_minus_max_zero0 (dB)
1	1	0.010961319	54.0	93.9
1	10	0.001350528	74.2	113.9
1	100	0.000097393	96.7	132.6
1	500	0.000008493	110.5	146.2
1	1000	0.000009353	116.5	152.1
10	1	0.011399652	71.7	111.4
100	1	0.011405154	91.7	131.4
500	1	0.011405208	105.7	145.4

alpha0	beta0	MIN min_hydro_minus_max_zero1 (dB)	MAX min_hydro_minus_max_zero1 (dB)	MAX |Far Field Difference| (dB)
1	1	43.8	89.2	4.30
1	10	63.8	107.6	n/a
1	100	84.2	125.2	n/a
1	500	97.9	138.9	0.95
1	1000	103.8	144.8	0.90
10	1	43.6	88.7	n/a
100	1	43.6	88.6	n/a
500	1	43.6	88.6	4.00

Table 1 – Solution Set Methodology Data Summary

VI. ERROR ANALYSIS – EFFECT OF POSITIONAL UNCERTAINTY

A. INTRODUCTION

This section addresses positional uncertainties in the placement of the near field array and its resulting effect on the extrapolated far field response using the described methodology. The examples provided herein were computed using the same device configurations, acoustic parameters and CHIEF set-ups detailed in the previous Solution Set using Methodology section. In this section, however, the alpha0 and beta0 terms were established at unity each and were not allowed to vary.

Positional uncertainty was defined to correspond to an inexact determination of range, vertical placement or angular orientation of the near field array with respect to the transmitting array's origin. The data for this section was obtained by varying one of the aforementioned parameters and then running through the computation steps in the methodology to achieve extrapolated far field response performance. The assessment of the error so induced is primarily based upon the examination of the far field responses and their differences. Near field responses (showing excellent agreement throughout the various error conditions) and figures of the $\mathbf{M}_{\mathbf{a\text{-}new}}$ matrices are provided for completeness in this section.

B. RANGE ERROR

Figures 63 through 86, inclusive, examine the effect of intentionally induced errors in range. For these figures, the simulated data used range values with error while the thesis technique used the nominal range magnitudes (ie: range without error). Various range errors were utilized to assist in evaluating the sensitivity of the technique to this parameter. These are summarized on the following page.

Figures 63 through 68 -2*lambda (-0.628 meters)

Figures 69 through 74 +2*lambda (+0.628 meters)

Figures 75 through 80 -1 cm

Figures 81 through 86 +1 cm

For reference, Figures 10 through 18 correspond to a range error equal to zero (and with alpha0=1 and beta0=1).

As expected, the larger range errors provided the most error in the extrapolated far field response as shown in the far field difference data reported in Figures 68 and 74. Even in these extreme cases (+/- 0.628 meters on a defined range of about 5.354 meters), however, the far field response (neglecting the response in the nulls of the response) differed from the zero range error case on average no more than about 0.5 dB. These results were a bit unexpected.

The range errors of +/- 1 cm were computed in this data set to better represent range errors that are more likely to be encountered within the test facility (APTF) [1] . These error conditions (Figures 74 and 80) show very little difference from the zero range error case. Thus, expected uncertainties in the range (to +/- 1 cm and for the set-up conditions examined) are not deemed to be a significant source of error in extrapolating far field response via the technique presented.

C. VERTICAL PLACEMENT ERROR

Figures 87 through 104, inclusive, examine the effect of intentionally induced errors in the vertical placement of the receive array. In like fashion as for the range error case, the simulated data used the error condition while the thesis technique used the

nominal values (ie: without error). Various errors were utilized to assist in evaluating the sensitivity of the technique to this parameter. These are summarized below:

Figures 87 through 92 +lambda/2 (+0.157 meters)

Figures 93 through 98 +lambda (+0.314 meters)

Figures 99 through 104 -lambda/2 (-0.157 meters)

The selection of these error values as multiples of lambda/2 stems from the configuration of the near field receive array itself. In the example that has been discussed, the spacing of the elements in the near field array is lambda/2. The worst error condition that was deemed likely to occur in practice was considered to be an offset in the placement of the array by one element. A 2 element offset (+lambda in this example) was selected to further assess the technique. This latter vertical error is, however, not likely to occur in practice for the inter-element spacings specified in this example. As for the range error case, Figures 10 through 18 can be referenced for comparison to a zero error case.

Again, as expected, the larger vertical placement errors provided the most error in the extrapolated far field response as shown in the difference data reported in Figure 98. For the more typical error cases (vertical errors of +/- lambda/2), the differences in far field response were comparable with those obtained for the zero error case. None of the vertical placement errors accounted for more than perhaps a 0.25 dB error in the final reported far field response (again, neglecting the response in the nulls of the response).

Thus, uncertainties in the vertical placement of the near field array (to within an element spacing and for the set-up conditions examined) are not deemed to be a significant source of error in extrapolating far field response via the technique presented.

D. ANGULAR ORIENTATION ERROR

Figures 105 through 110, inclusive, examine the effect of intentionally induced errors in the angular orientation (offset) of the receive array with respect to the transmitting array. In like fashion as for the previous two cases, the simulated data used the error condition while the thesis technique used the nominal values (ie: without error). A single value of error equivalent to a 10 degree offset was examined in this data set.

Figure 110 shows the difference between the simulated data set and that obtained from the thesis technique. Upon initial inspection, this figure seems to indicate gross errors caused by the 10 degree displacement. Differences of close to 30 dB are noted in this figure. Upon closer examination of the defining constituents of the difference figure (the data presented in Figure109), it is readily seen that the shape and structure of the two far field responses are essentially the same. The introduction of the 10 degree angular error has shifted the response accordingly and caused the large difference errors. While this results in large difference errors (Figure 110), this condition is easily post-processed and does not represent any substantial source of error or a deterrent to the use of the thesis methodology.

Thus, as for the other cases, uncertainties in the angular orientation of the near field array are not deemed to be a significant source of error in extrapolating far field response via the technique presented. This statement is likely to hold true for all set-ups and not just for the unique parameters looked at in this example set.

E. ERROR SUMMARY COMMENTS

Table 2 summarizes the error conditions examined in the previous examples in this section. Please reference the previous section, Solution Set using Methodology, for definitions of the nomenclatures utilized.

For the given example set-up, errors in range, vertical placement and angular orientation of the near field receive array (when taken singularly and not taken in any kind of an extreme sense) induced very little error in the extrapolated far field response (again, neglecting the nulls of the response) Attention to error quantification in terms of some lambda value (or fractional value) was given in an effort to assist in extrapolating this performance data to other set-up conditions. While this ability to extrapolate performance traits is likely achievable, the ability to position devices within the same lambda tolerance will be hampered as operating frequencies increase (and wavelengths correspondingly decrease).

The overall conclusion provided, however, is that the technique worked extremely well at characterizing far field response even when physical set-up parameters were known to be in error.

alpha0=beta0=1	MAX |Near Field Hydro Diff| (dB)	MIN min_hydro_minus_max_zero0 (dB)	MAX min_hydro_minus_max_zero0 (dB)
error condition			
no error	0.010961319	53.95	93.86
range error: -2*lambda	0.015705251	49.88	88.29
range error: +2*lambda	0.010182845	53.68	93.62
range error: -1 cm	0.011011699	53.96	93.78
range error: +1 cm	0.010911516	53.95	93.93
vert rec array error: +lambda/2	0.019242789	50.60	89.97
vert rec array error: +lambda	0.023584640	40.73	85.23
vert rec array error: -lambda/2	0.019242789	50.60	89.97
azimuthal error: +10 degrees	0.011221058	53.95	93.86

	MIN min_hydro_minus_max_zero1 (dB)	MAX min_hydro_minus_max_zero1 (dB)	MAX |Far Field Difference| (dB)
error condition			
no error	43.80	89.16	4.30
range error: -2*lambda	40.43	83.27	12.60
range error: +2*lambda	41.85	92.73	5.20
range error: -1 cm	43.83	89.05	4.40
range error: +1 cm	43.78	89.26	4.25
vert rec array error: +lambda/2	35.09	87.88	3.00
vert rec array error: +lambda	29.10	81.83	2.60
vert rec array error: -lambda/2	35.09	87.88	3.00
azimuthal error: +10 degrees	43.80	89.16	30.00

Table 2 – Effect of Positional Uncertainty Data Summary

VII. ADDITIONAL SOLUTION SETS

A. INTRODUCTION

This section addresses the effectiveness of the technique when the near field array is placed closer to the device under test. The same transmitting array, 1.885 meters in length and operating at a frequency of about 4775 Hz, is used in all cases. The cases presented in this section are summarized below:

	Near Field Array Location	alpha0	beta0
Figures 111 to 113	2.082 meters	1	1000
Figures 114 to 116	1.466 meters	1	1000
Figures 117 to 119	1.466 meters	1	10,000
*Figures 120 to 122	1.466 meters	1	1000

* − Figures 120 to 122 represent data taken with a different configuration near field array. The array used for this data set was about 2.199 meters long and was composed of 15 equally spaced elements.

The data presented here was provided more for academic reasons than for any required need. The location of the near field array at a range of 5.354 meters in the previous sections adequately met the restrictions imposed by the dimensions of the APTF test vessel. Thus, the test set-ups specified in this section may not be required, but they do serve to address the capabilities of the technique.

B. NEAR-FIELD ARRAY RANGE OF 2.082 METERS

Figures 111 through 113 used a near field array range of 2.082 meters. The near field data sets (simulated and from the technique) matched very well; the far field data

sets also matched extremely well. It is interesting to note that the near field data actually now shows a local null at the same angular location where the far field data actually peaks.

Please note that the enclosing sphere required for these data sets was also moved to a closer proximity to the device under test. Since the radius of this enclosing sphere was decreased and yet the frequency of operation stayed the same, fewer numbers of patches were required in the vertical direction and a lower symmetry order sufficed for the data set. The lower symmetry order (since it is directly related to the number of discrete angles in the responses) accounts for the choppiness of these responses when compared to the previous solution sets' figures.

C. NEAR-FIELD ARRAY RANGE OF 1.466 METERS

Figures 114 to 122 all used a near field array range of 1.466 meters. Figures 114 through 116 and Figures 117 through 119 used the same alpha0 values (one) but differing beta0 values of 1000 and 10,000, respectively. The far field response of Figure 115 showed a very reasonable fit to the simulated data along the main beam of the device, but did not have as good a match in the side lobes. The data for Figure 118 was computed with a higher beta0 value to attempt to correct this side lobe problem. The $\mathbf{M}_{\mathbf{a\text{-}new}}$ data for these two conditions (Figures 116 and 119) would lead one to believe that the higher beta0 value would give a more accurate description of the devices actual far field response. This was not found to be the case. The far field response for these two conditions (Figures 115 and 118) looked to be very comparable.

After some experimentation, it was observed that increasing the length of the near field array yielded an improvement in side lobe performance of the device under test. Figures 120 to 122 show response with a near field array that has been lengthened by 2*lambda to 2.199 meters (originally this array was about 1.571 meters in length). The far field response (Figure 121) shows an improved side lobe performance over the previous data sets using the same near field array range of 1.466 meters.

Please note that the number of elements in the 2.199 meter near field array was increased from 11 to 15 in an effort to maintain lambda/2 spacing. The improvement in far field response observed, however, was due to its length increase and not due to the increase in the number of its comprising elements. This latter characteristic was arrived at empirically but is not supported with any data in this write-up.

D. SUMMARY

Based upon the data sets presented in this section, it is desirable to quantify a minimum near field array length and a minimum test distance for a given test set-up to provide some assurance that proper far field performance is extrapolated. Using empirical results, it is believed that a near field array with a length that exceeds the maximum dimension of the device under test placed at a separation distance comparable to its length could achieve this condition. This latter hypothesis is speculated but is not stated as proof in this thesis. It is however, supported by the data presented in Figures 123 through 125 (with the transmitting array, near field array and range all specified at 1.885meters).

THIS PAGE INTENTIONALLY LEFT BLANK

VIII. SUMMARY

A. INTRODUCTION

A technique to extrapolate far field performance of an acoustic device from a data set obtained in the near field was demonstrated. The technique successfully married the Helmholtz integral relationships (**A** and **B** matrices and pressure computations utilizing the integration of the Green's function and its gradient) and expected physical parameters with near field acquired data to yield an over determined system. Solution via a least squares technique provided pressure and normal velocity terms for each patch of the enclosing sphere surface. This characterization of the enclosing sphere's surface allowed extrapolation to any range (near or far field) when Greens' function and its gradient integration values appropriate to that range are utilized.

B. SOLUTION SETS

An example solution set was provided that demonstrated the ability of the technique to produce extremely accurate results. Figures 4 and 5 below show the data computed for the near field and far field, respectively, for the defined acoustic device (a 1.885 meter long linear array operating at about 4775 Hz). (Figures 4 and 5 represent the same data set shown in Figures 34 and 36.)

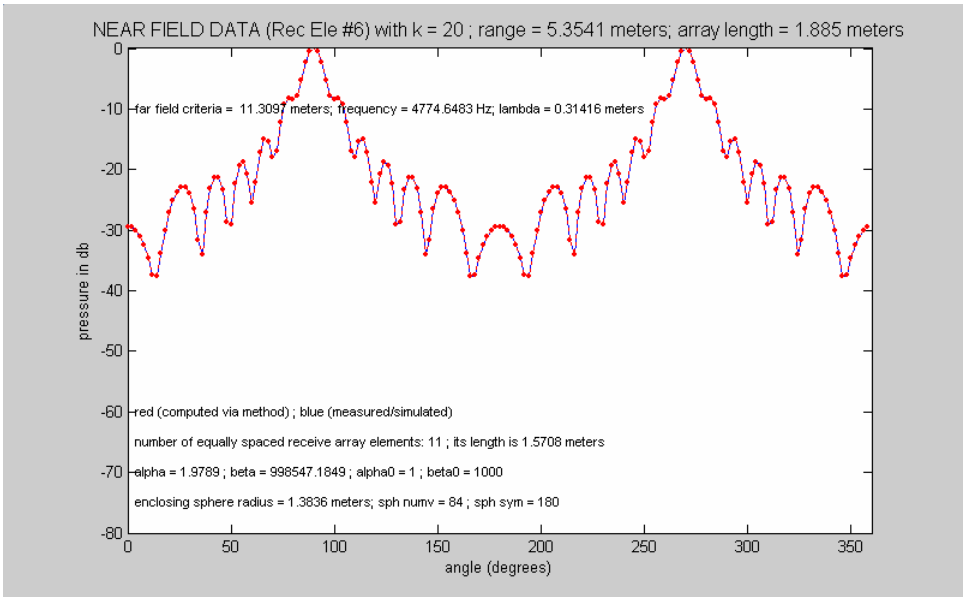


Figure 5 – Near Field Data – computed and simulated

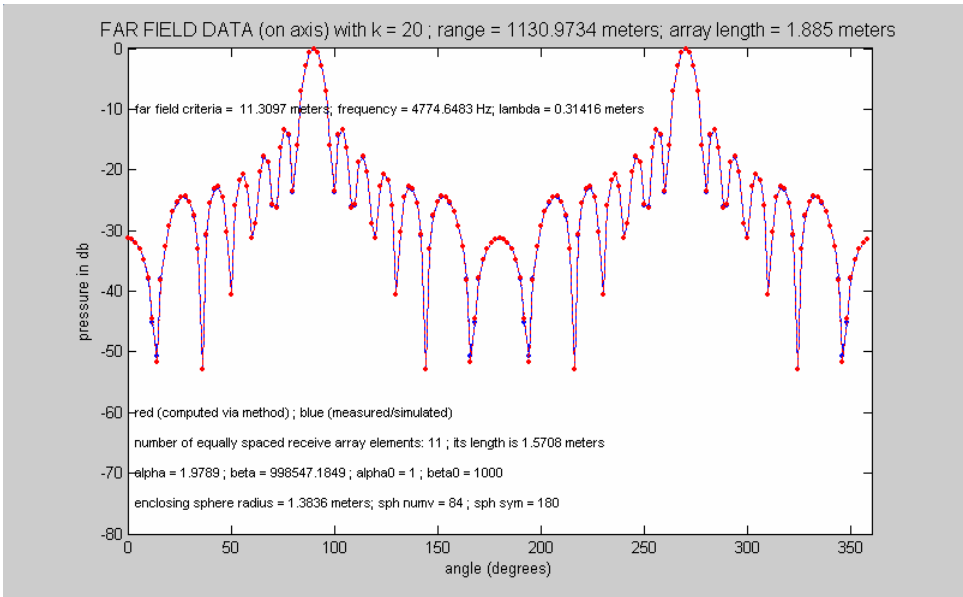


Figure 6 – Far Field Data – computed and simulated

The example set-up was specifically designed for a minimal set of elements in the near field array (eleven). This minimal set was utilized in an attempt to ease the adaptation of the technique to a future hardware implementation (ie: 16 channel acquisition cards of sufficient dynamic range and frequency of operation are readily

available now; the eleven near field array would require only a single card for implementation.)

C. SUMMARY

It was desired to verify the technique with actual measurement data, however, the present configuration of the Acoustic Pressure Tank Facility's measurement system (4 channel ; VXI-based) precluded a reasonably fast or convenient way to measure the complex response of an eleven element near field array. This latter array also does not presently exist, but likely could have been conjured up within existing USRD inventories if so required. It is anticipated that this technique will be looked at with an updated acquisition system and perhaps a specially designed array at some future date.

An error study was undertaken with the simulated data sets. This error study examined errors in range, vertical placement and angular orientation of the near field array. This study confirmed that the technique (at least for the given set-up) was extremely robust and did not require exacting tolerances to achieve qualifiable results. This was especially gratifying for the test environment in which the technique is ultimately intended to be used.

The technique will not, however, be appropriate for all devices and all operating frequencies. The reliance on COTS computing hardware and commercial software products does limit the size of the matrices that can be reasonably handled. As mentioned previously, the solution set example used a relatively low frequency of about 4775 Hz but yet still required the solution of 30,240 "unknowns" for its 15,120 individual patches.

In summary, the technique does present itself favorably and is highly likely to be looked at for future implementation at USRD.

THIS PAGE INTENTIONALLY LEFT BLANK

FAR FIELD EXTRAPOLATION TECHNIQUE USING CHIEF ENCLOSING SPHERE DEDUCED PRESSURES AND VELOCITIES

FIGURES

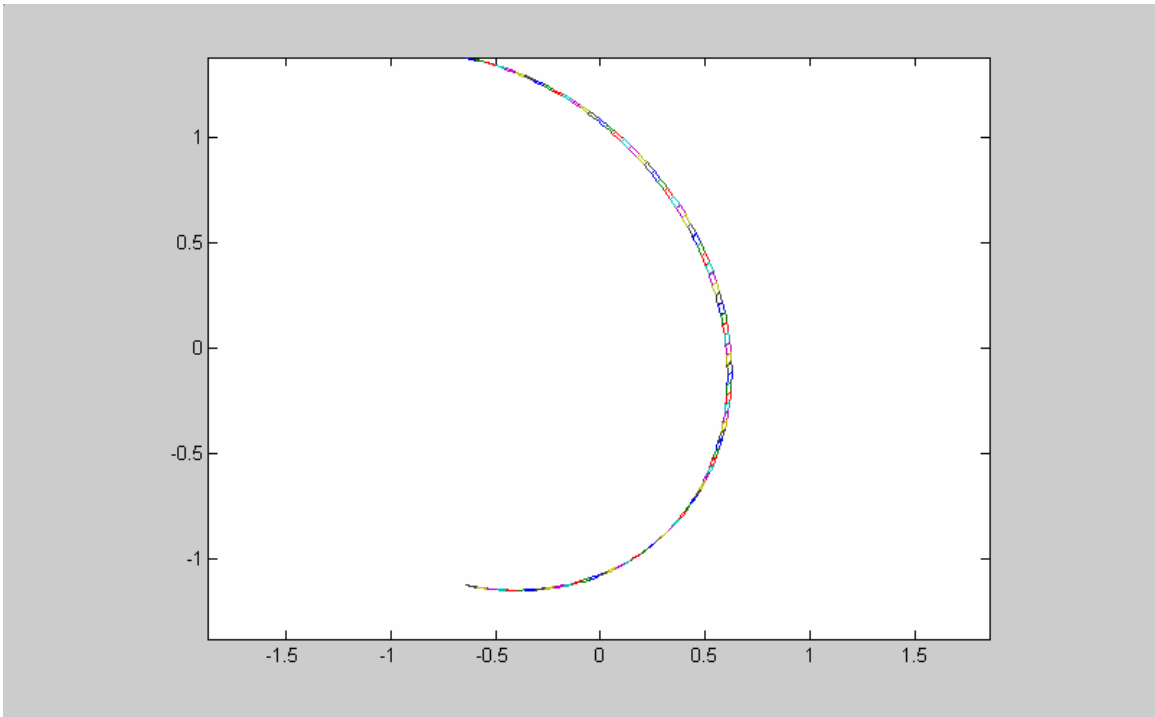


Figure 7 – Enclosing Sphere Section (numv =84; symmetry=180)

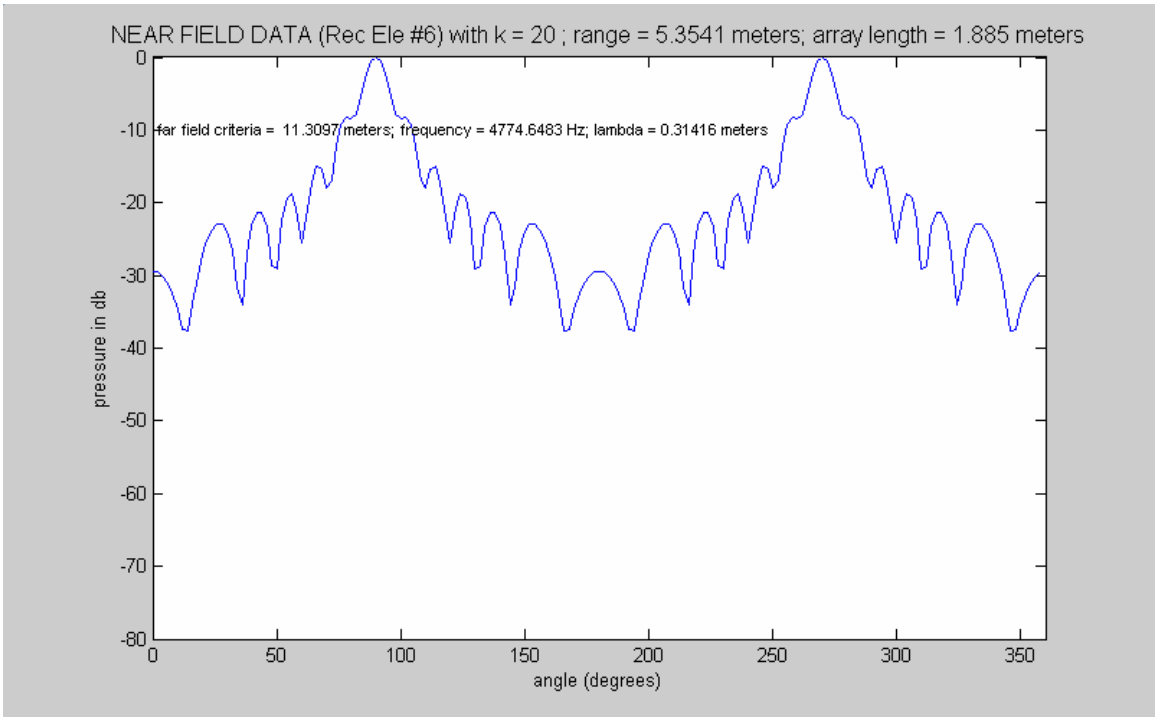


Figure 8 – Theoretical Near Field Response of Linear Array

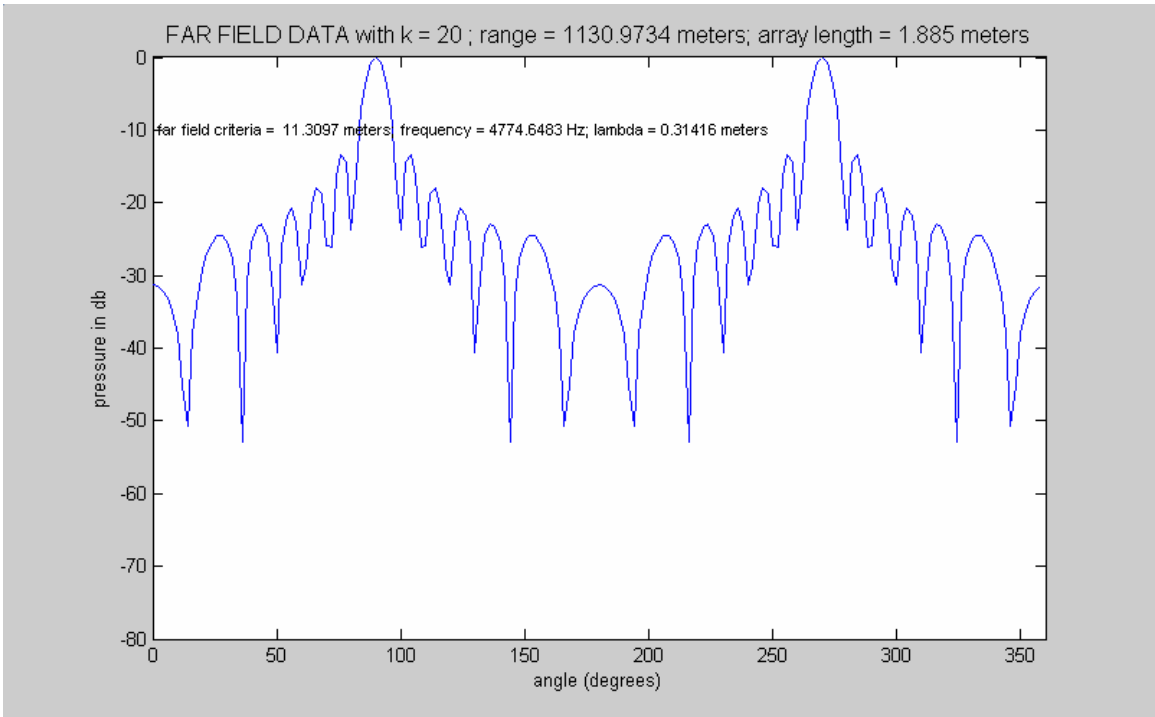


Figure 9 - Theoretical Far Field Response of Linear Array

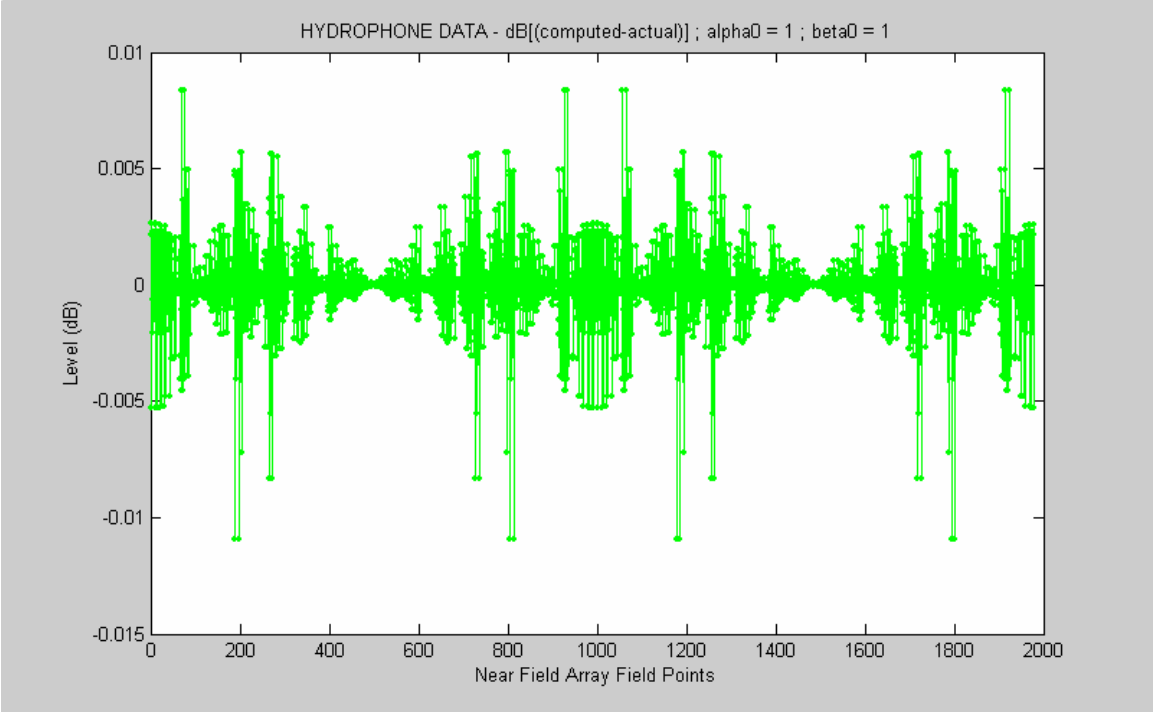


Figure 10 – Near Field Hydrophone Difference Data
alpha0=1; beta0 = 1

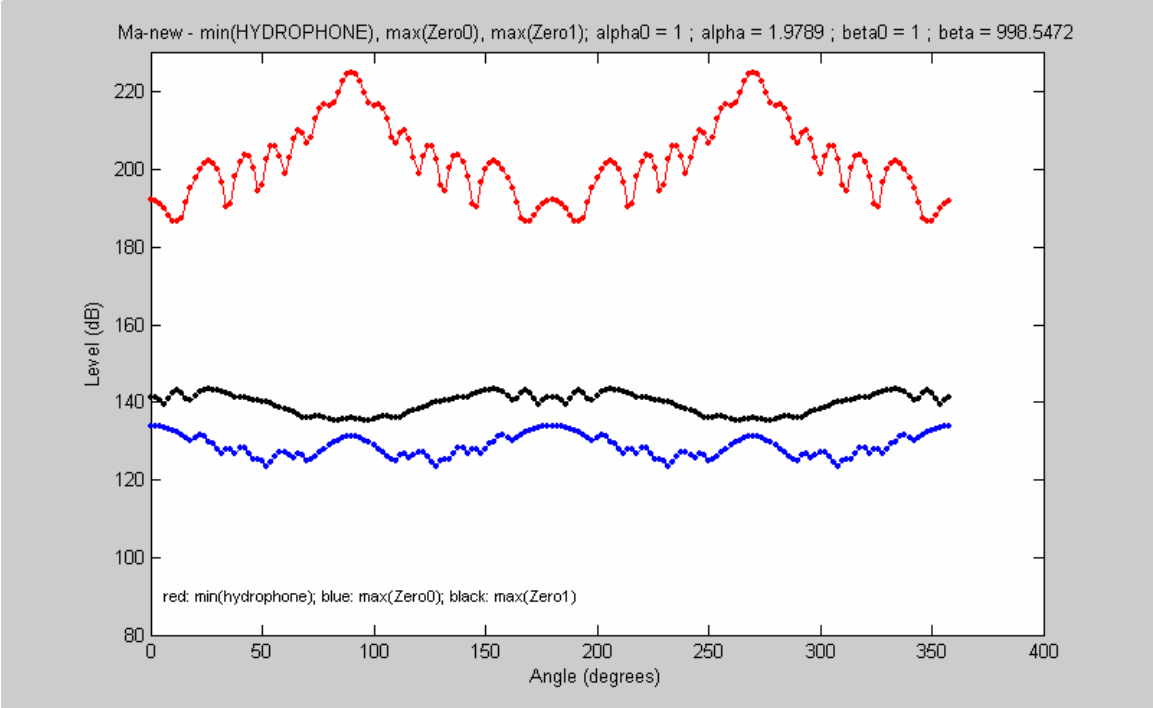


Figure 11 – $\mathbf{M}_{\mathbf{a\text{-}new}}$ Components
alpha0=1; beta0 = 1

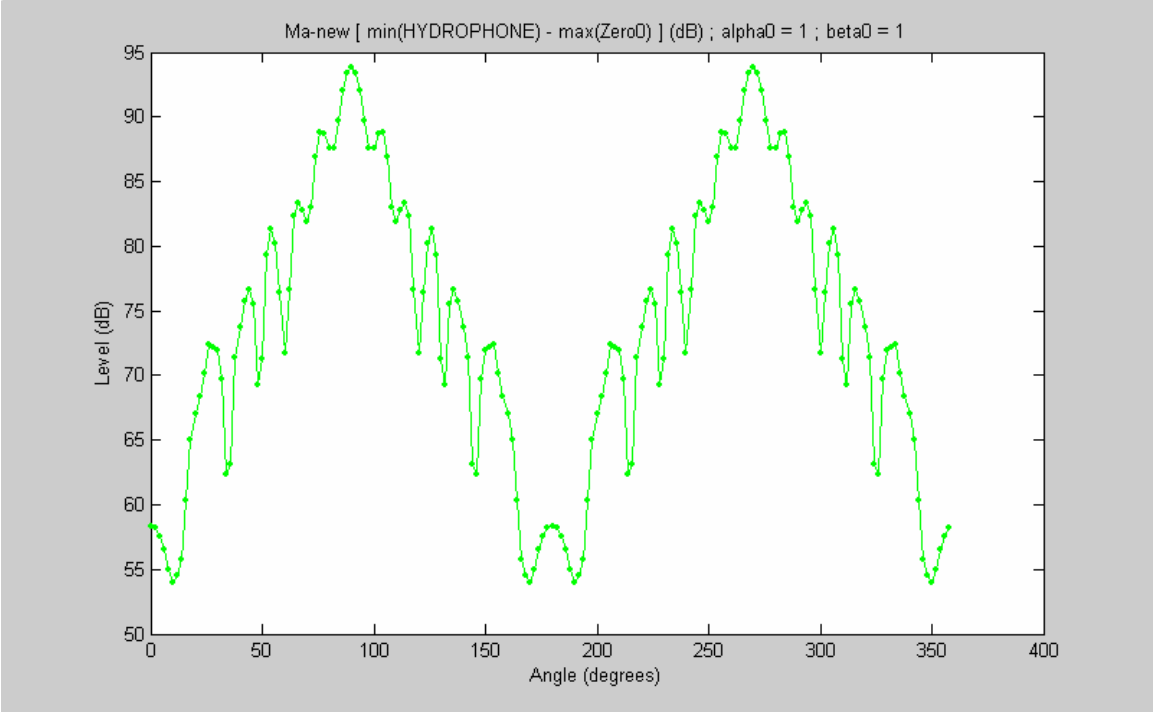


Figure 12 – $\mathbf{M}_{\mathbf{a\text{-}new}}$ [min(Hydrophone) – max(Zero0)] Data
alpha0=1; beta0 = 1

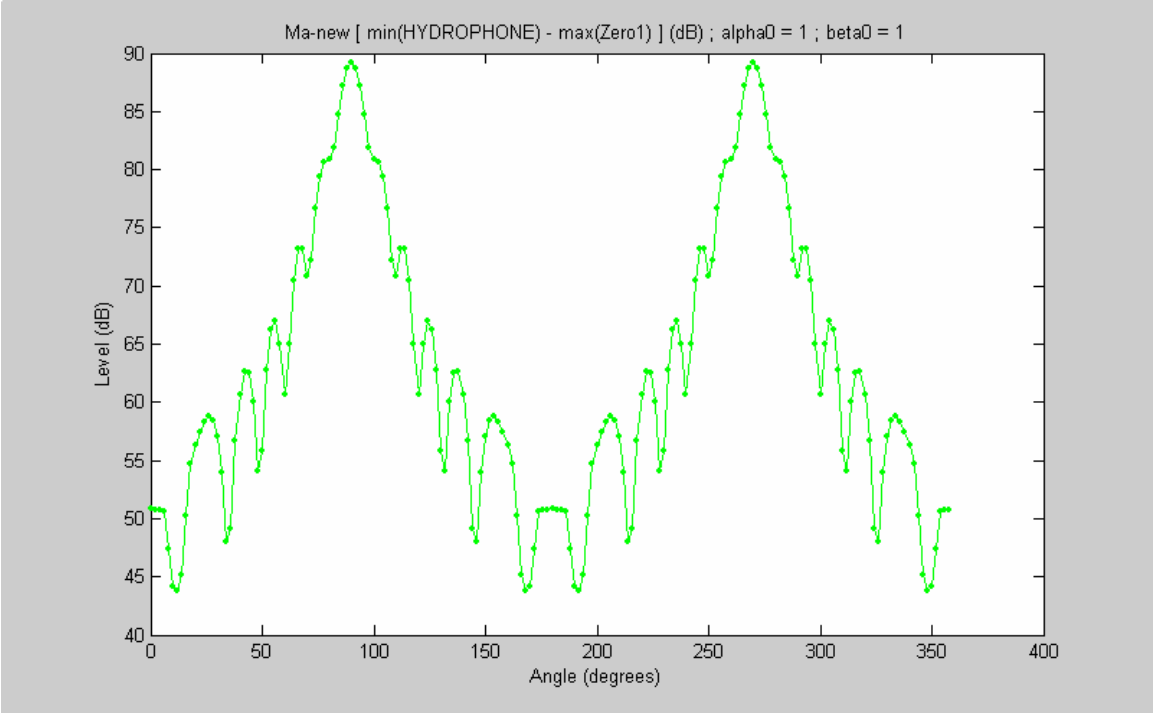


Figure 13 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ [min(Hydrophone) – max(Zero1)] Data
alpha0=1; beta0 = 1

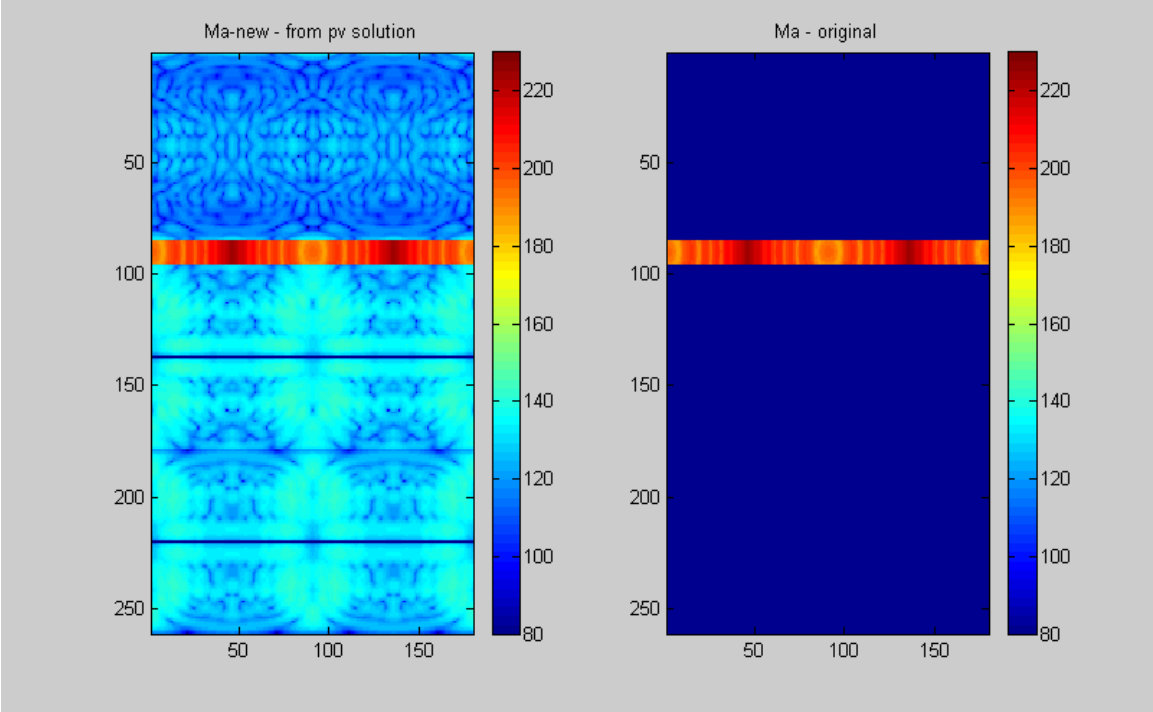


Figure 14 – $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0 = 1

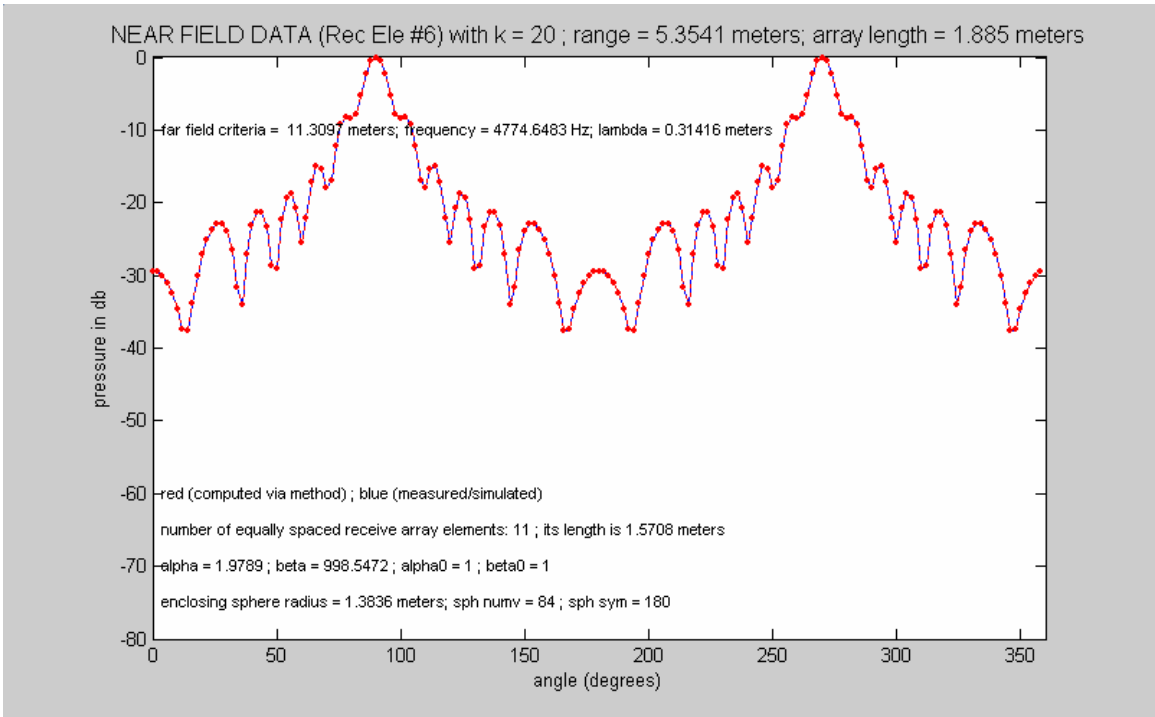


Figure 15 – Near Field Data – computed and simulated
alpha0=1; beta0 = 1

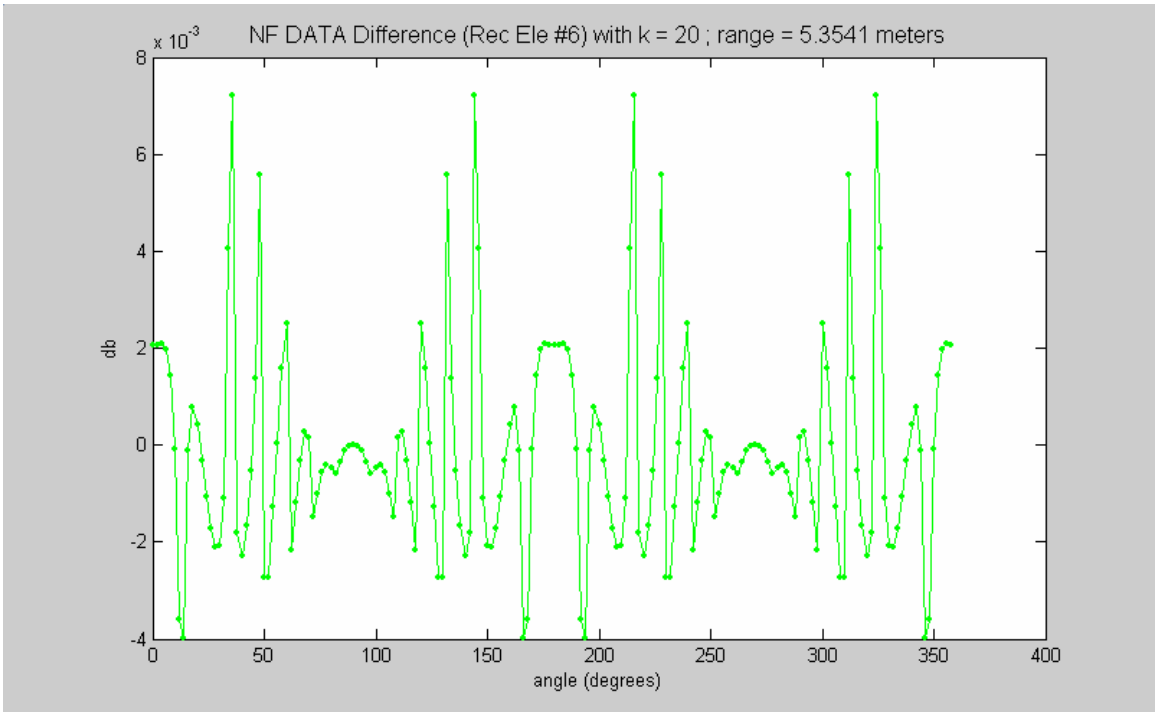


Figure 16 – Near Field Data Difference
alpha0=1; beta0 = 1

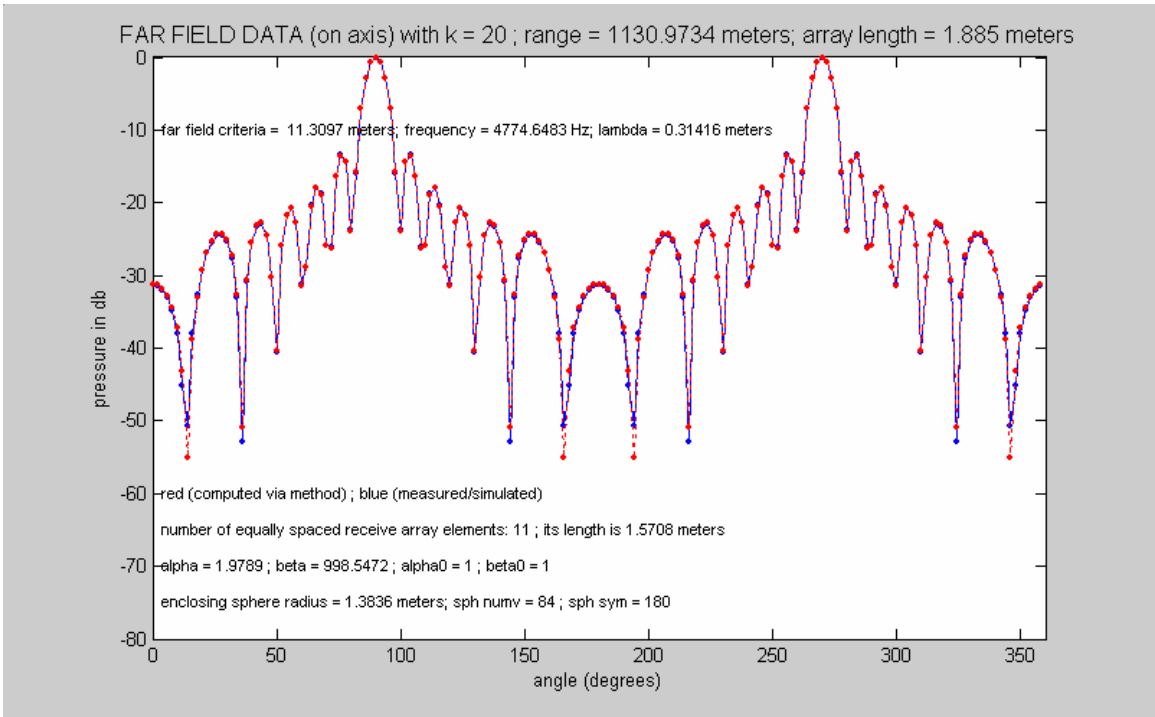


Figure 17 - Far Field Data – computed and simulated
alpha0=1; beta0 = 1

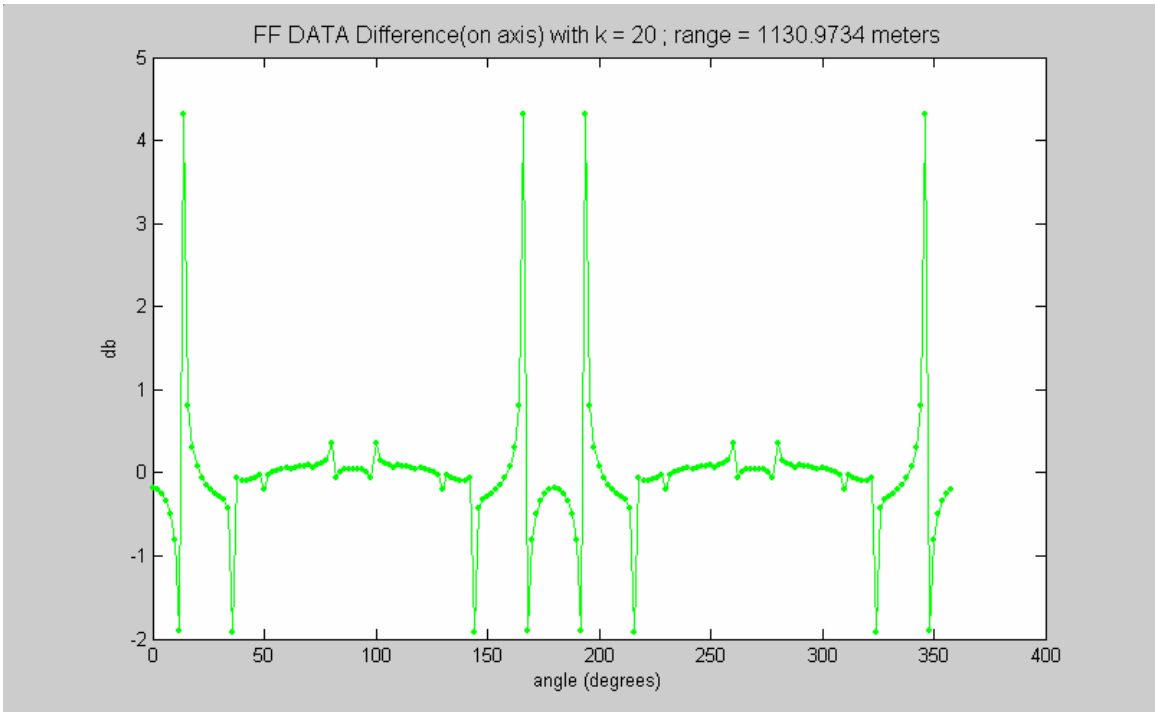


Figure 18 - Far Field Data Difference
alpha0=1; beta0 = 1

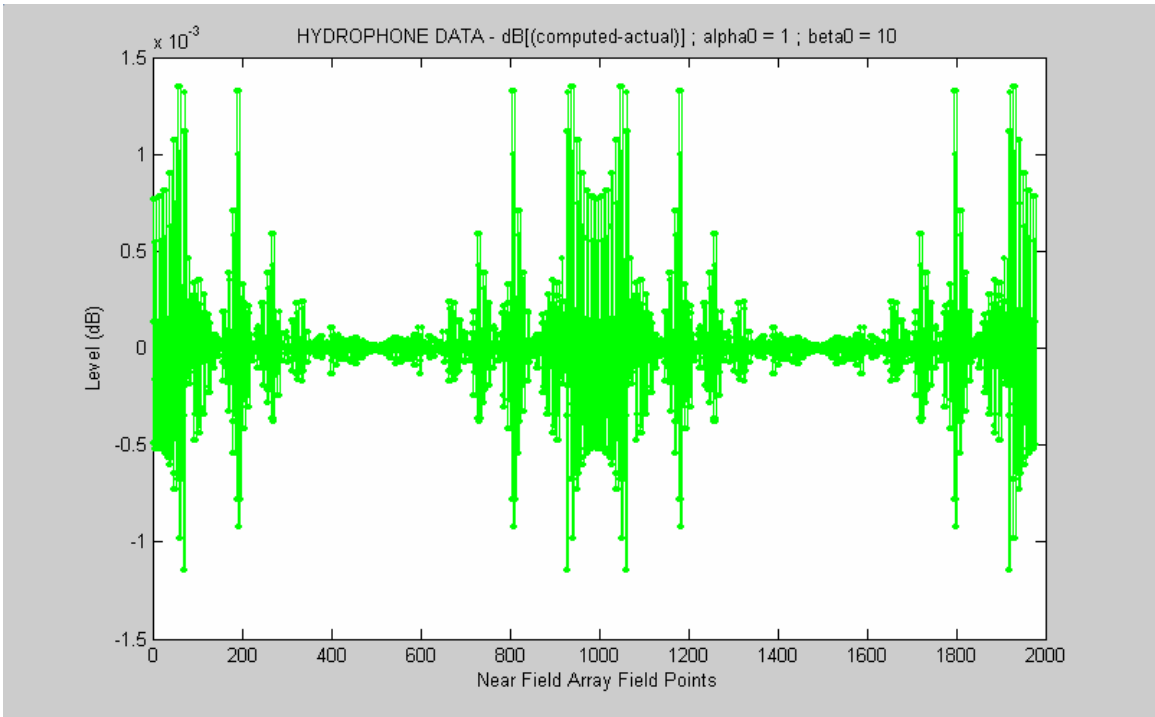


Figure 19 – Near Field Hydrophone Difference Data
alpha0=1; beta0 = 10

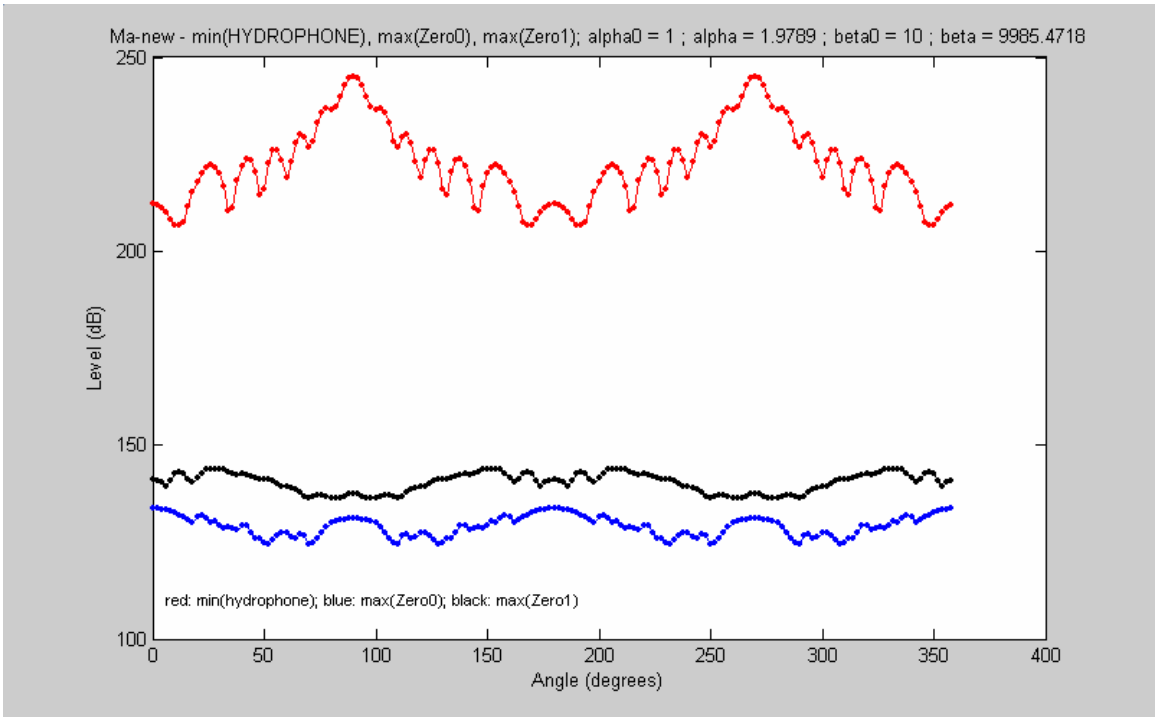


Figure 20 – $\mathbf{M}_{\mathbf{a\text{-}new}}$ Components
alpha0=1; beta0 = 10

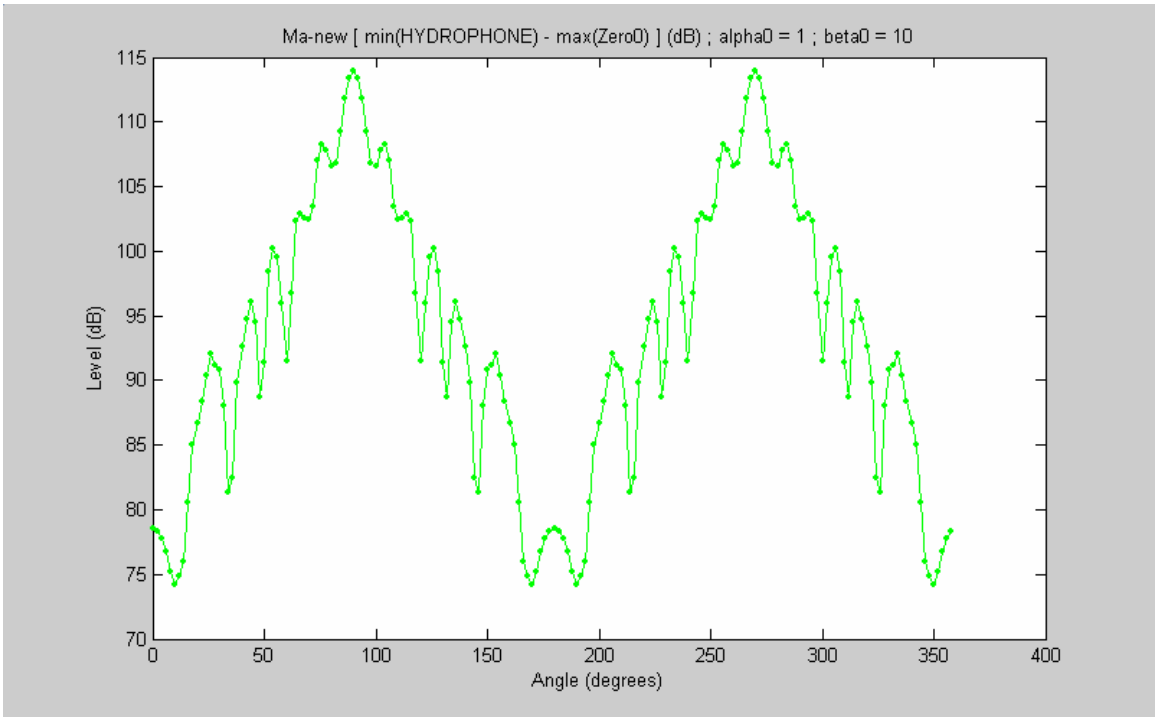


Figure 21 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ [min(Hydrophone) – max(Zero0)] Data
alpha0=1; beta0 = 10

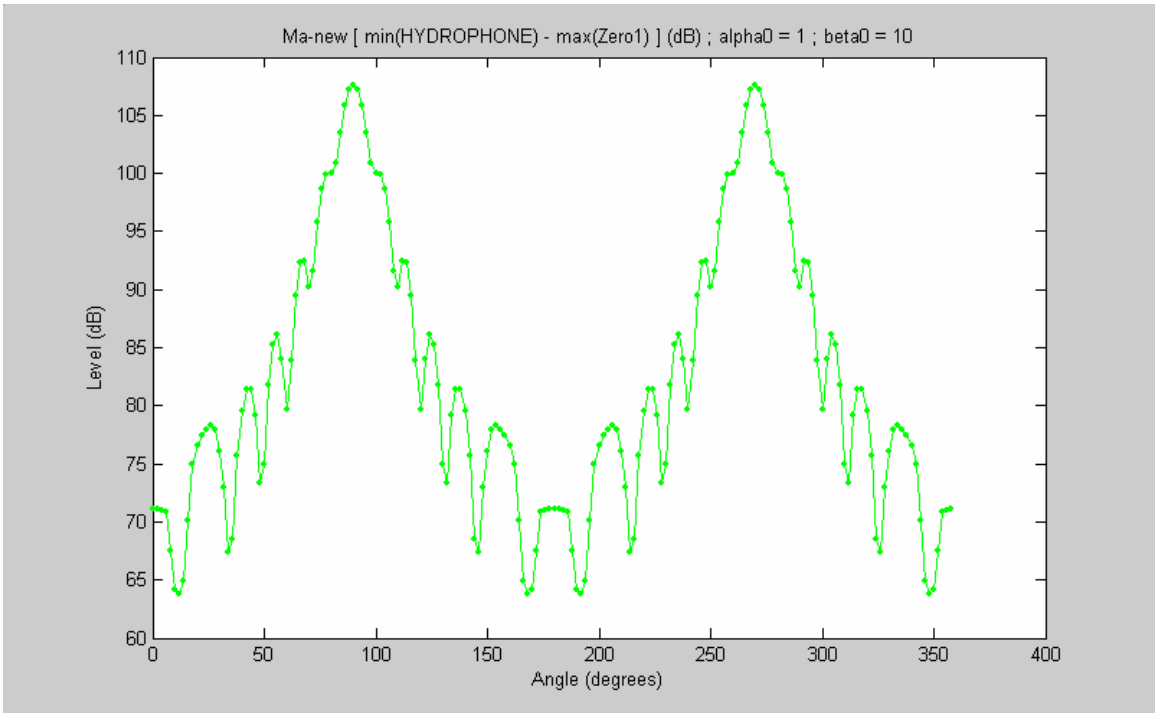


Figure 22 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ [min(Hydrophone) – max(Zero1)] Data
alpha0=1; beta0 = 10

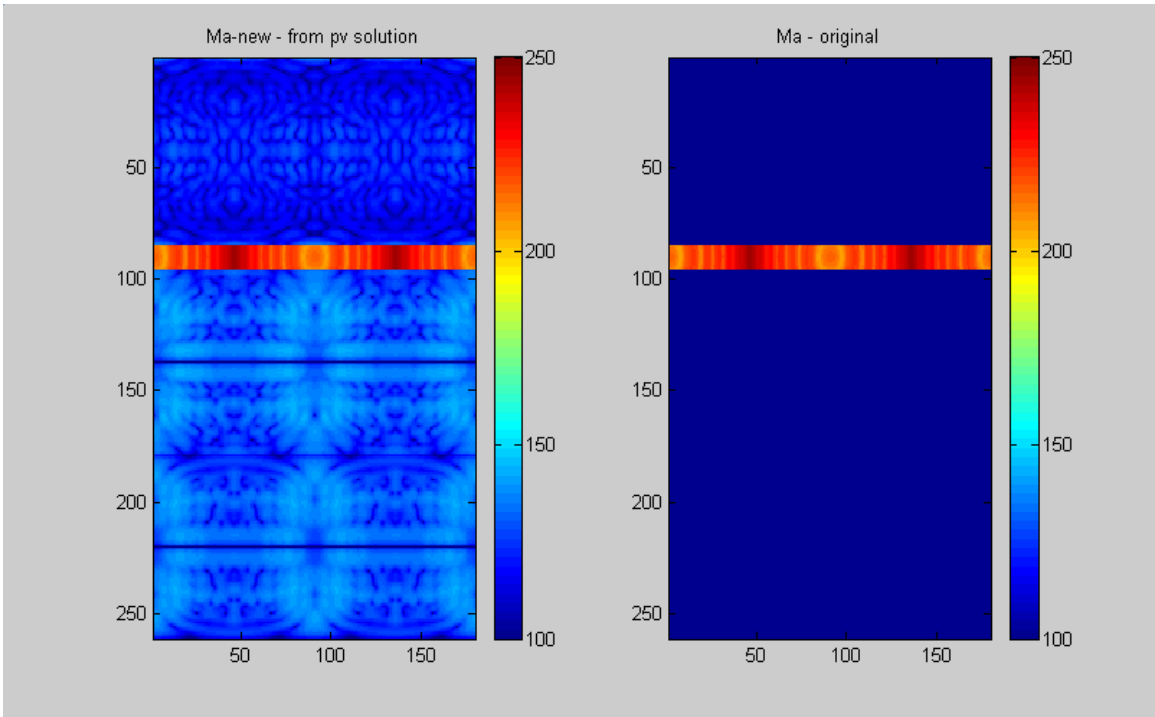


Figure 23 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0 = 10

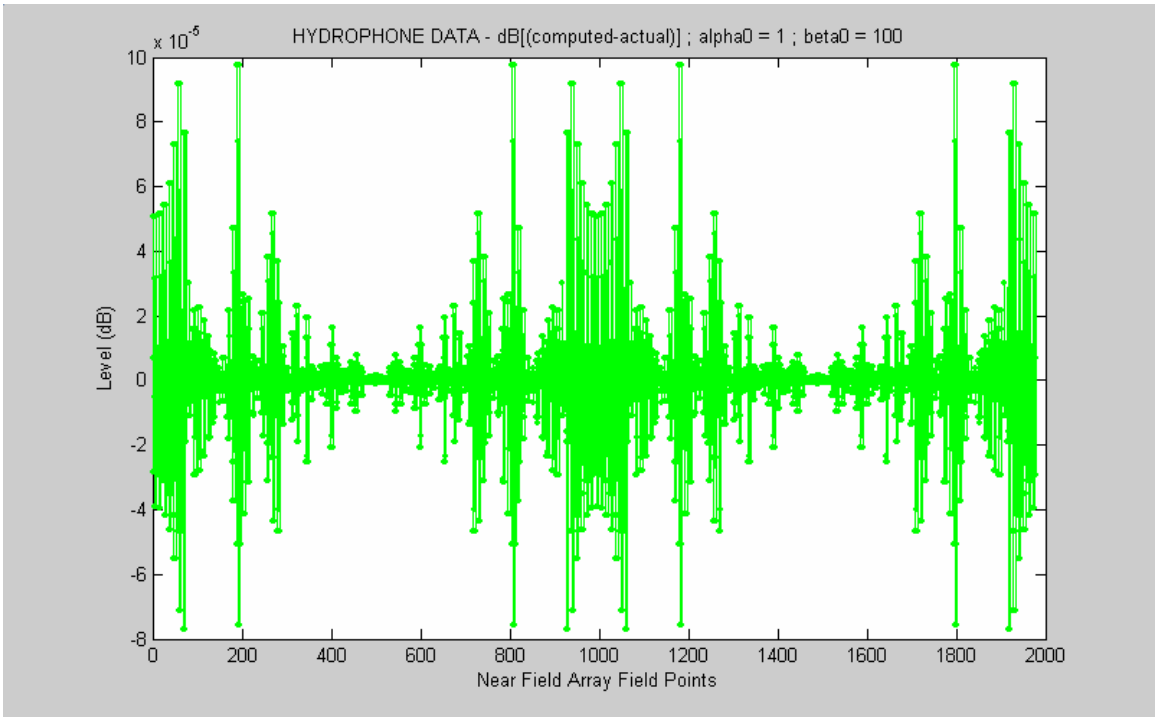


Figure 24 - Near Field Hydrophone Difference Data
alpha0=1; beta0 = 100

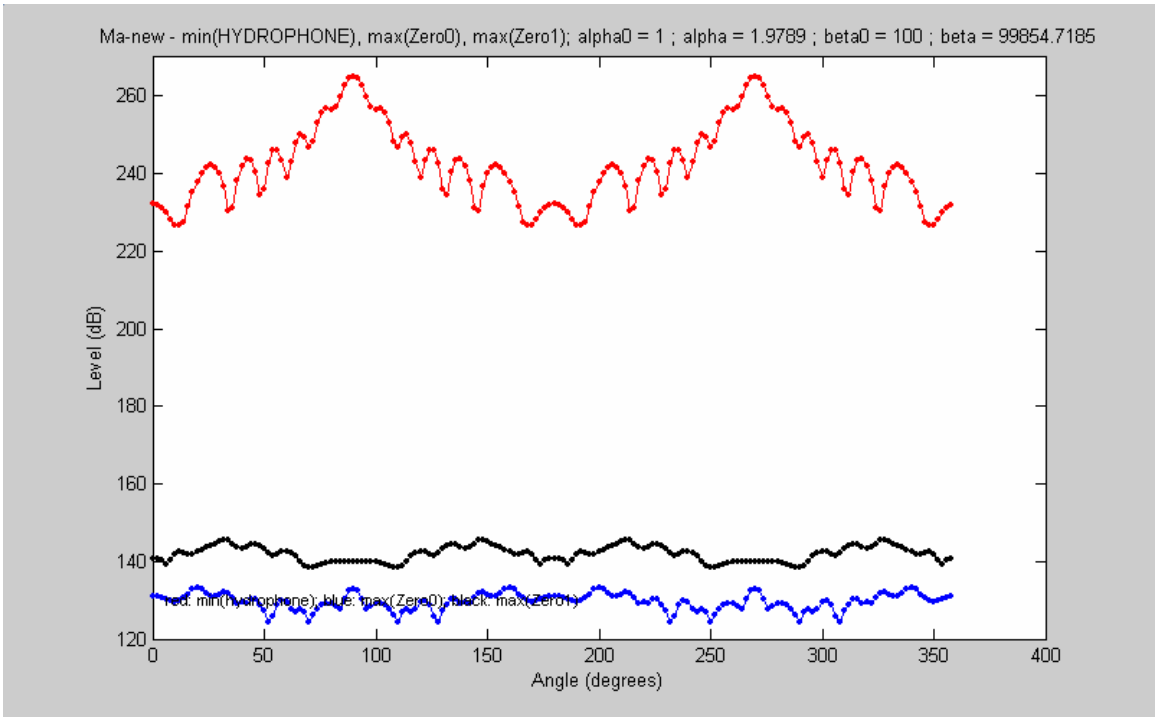


Figure 25 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ Components
alpha0=1; beta0 = 100

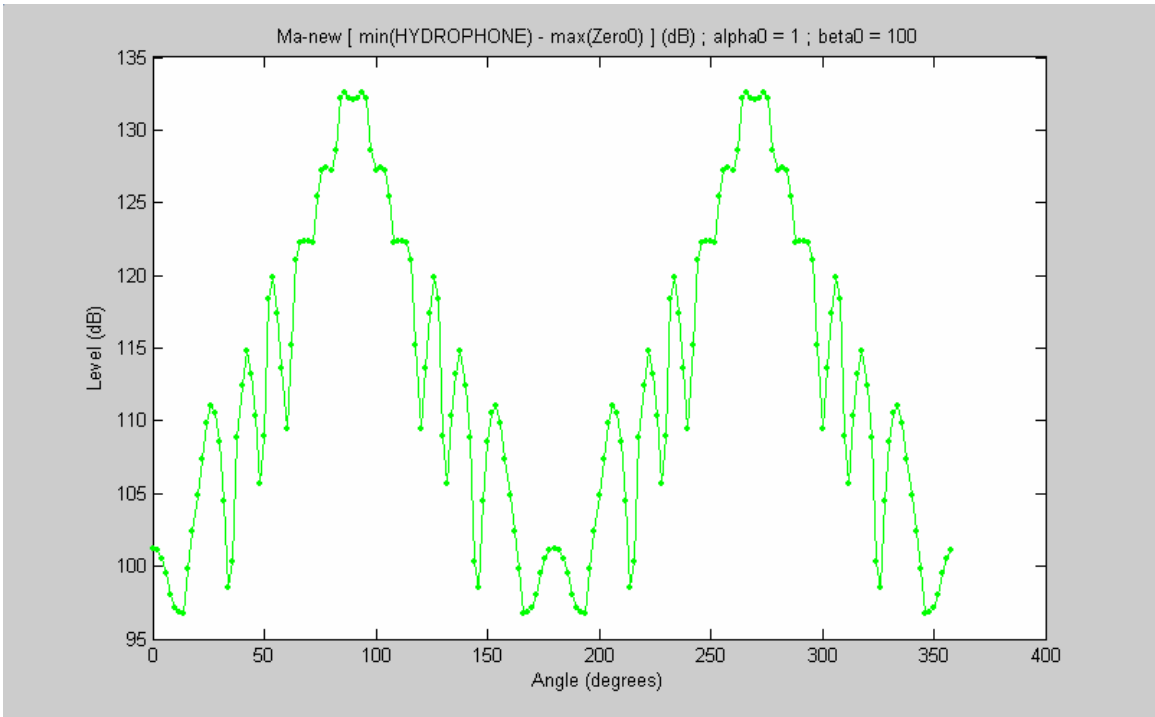


Figure 26 - $\mathbf{M_{a\text{-}new}}$ [min(Hydrophone) – max(Zero0)] Data
alpha0=1; beta0 = 100

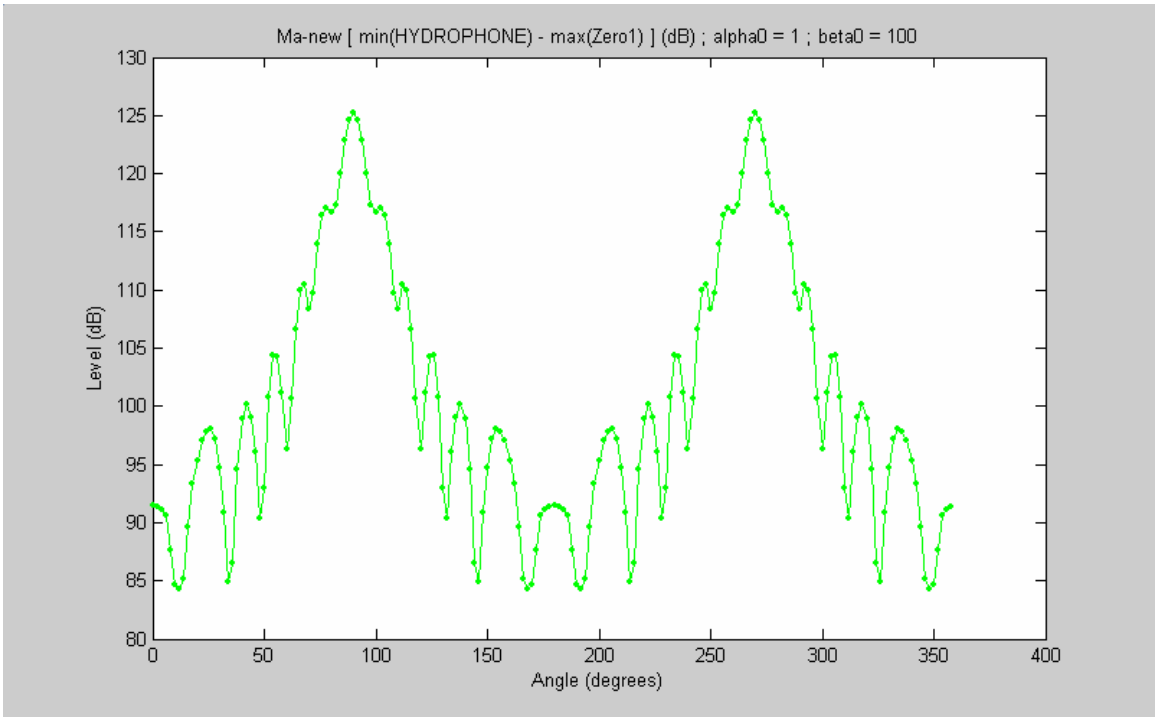


Figure 27 - $\mathbf{M_{a\text{-}new}}$ [min(Hydrophone) – max(Zero1)] Data
alpha0=1; beta0 = 100

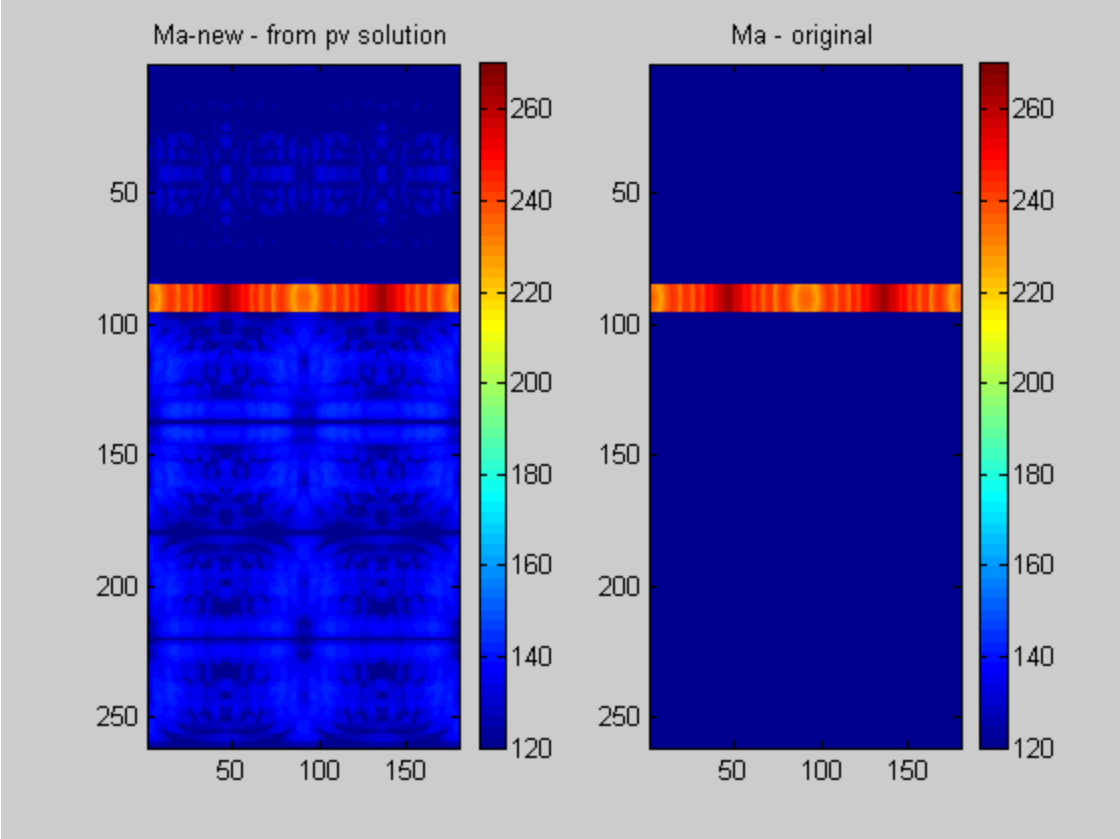


Figure 28 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0 = 100

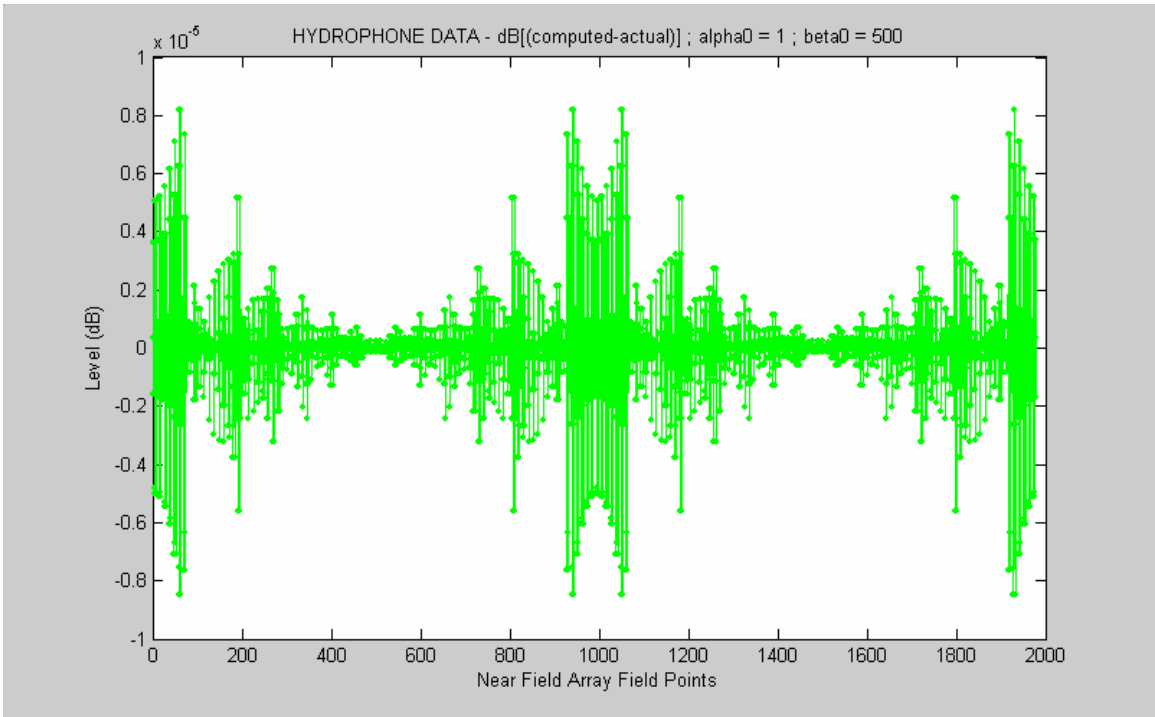


Figure 29 - Near Field Hydrophone Difference Data
alpha0=1; beta0 = 500

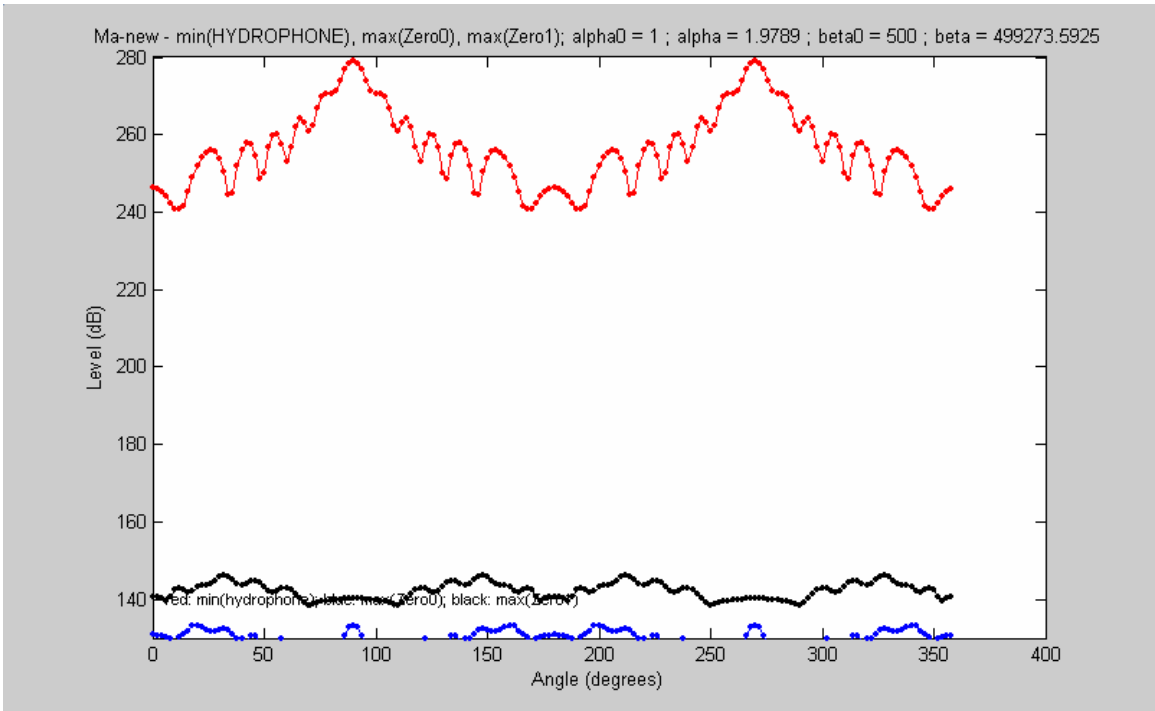


Figure 30 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ Components
alpha0=1; beta0 = 500

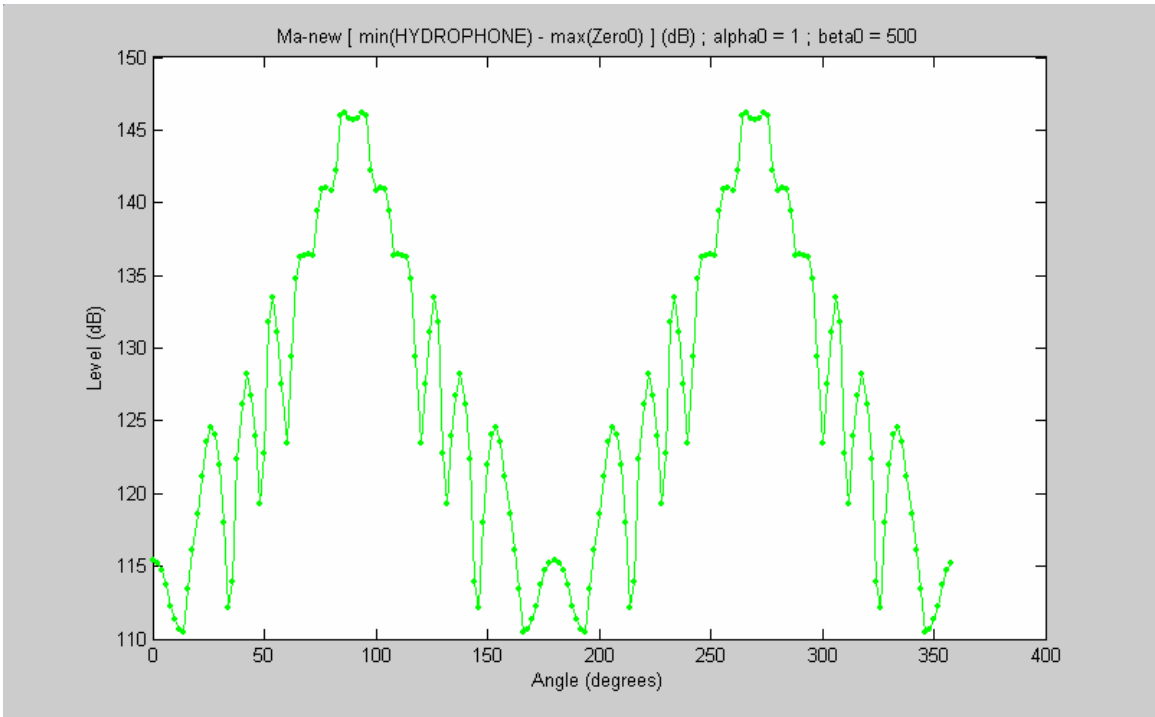


Figure 31 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ [min(Hydrophone) – max(Zero0)] Data
alpha0=1; beta0 = 500

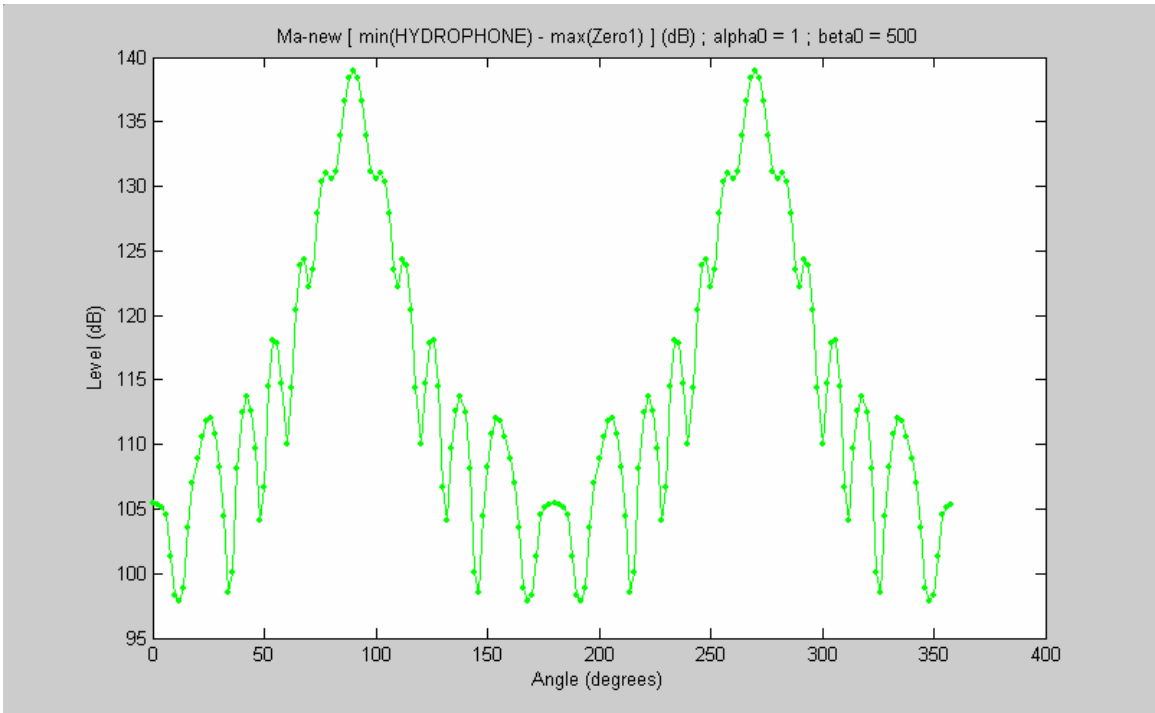


Figure 32 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ [min(Hydrophone) – max(Zero1)] Data
alpha0=1; beta0 = 500

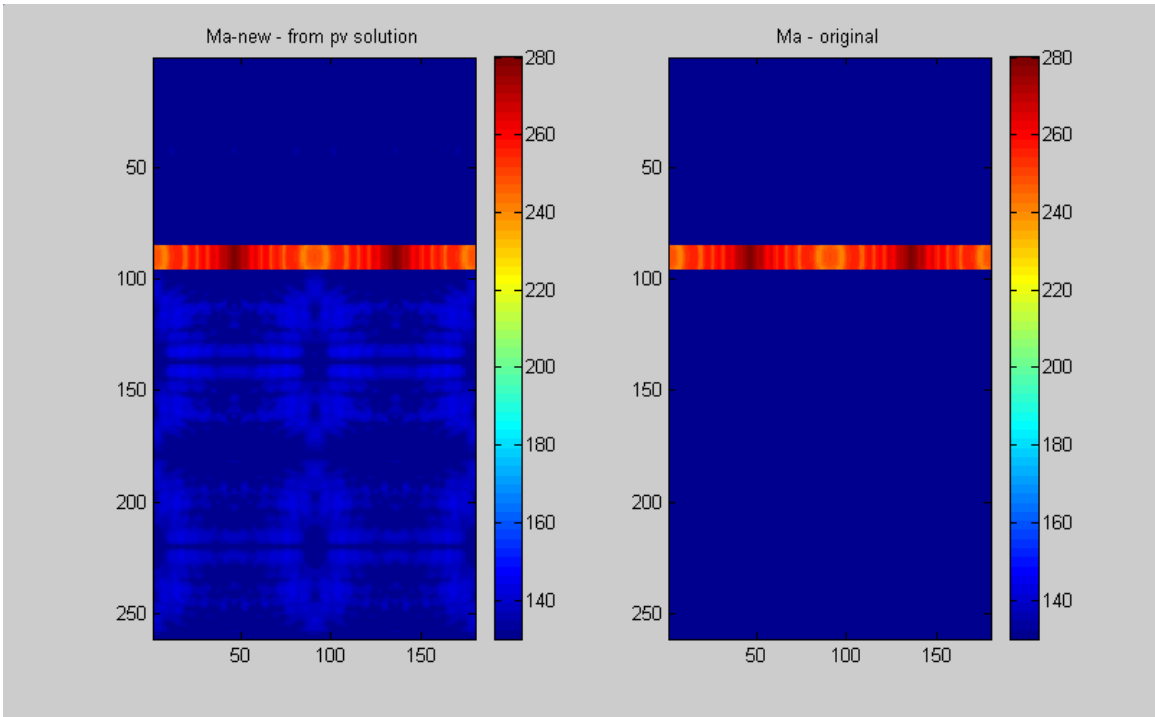


Figure 33 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0 = 500

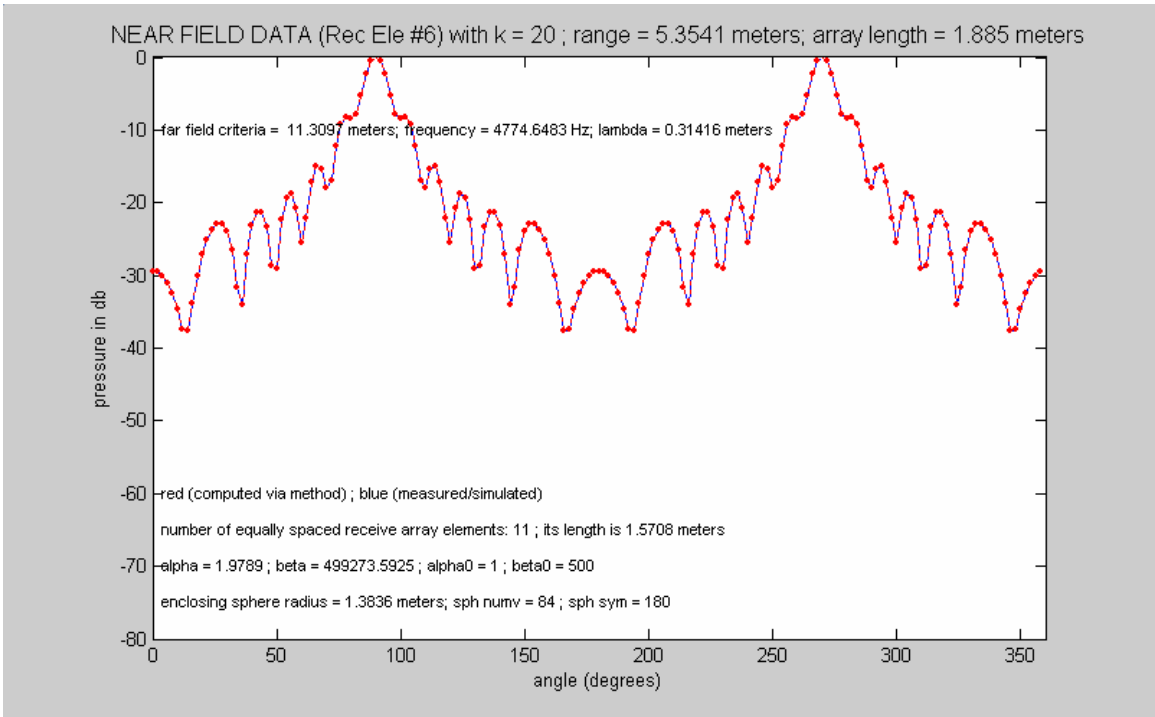


Figure 34 - Near Field Data – computed and simulated
alpha0=1; beta0 = 500

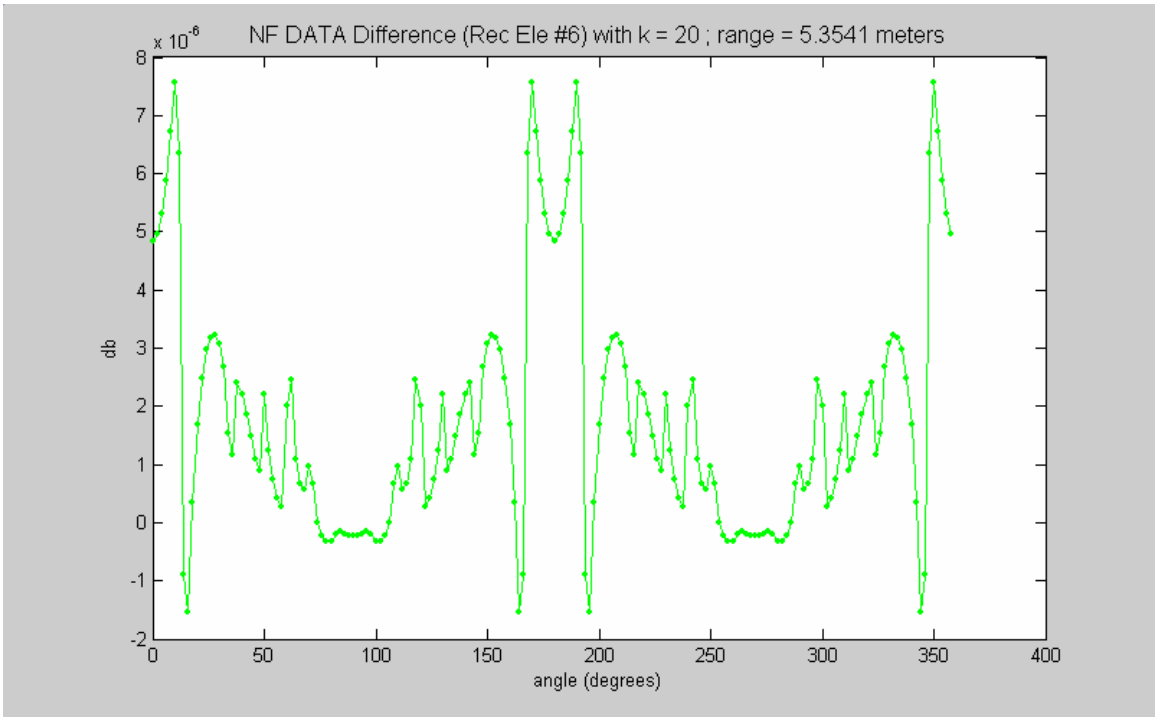


Figure 35 - Near Field Data Difference
alpha0=1; beta0 = 500

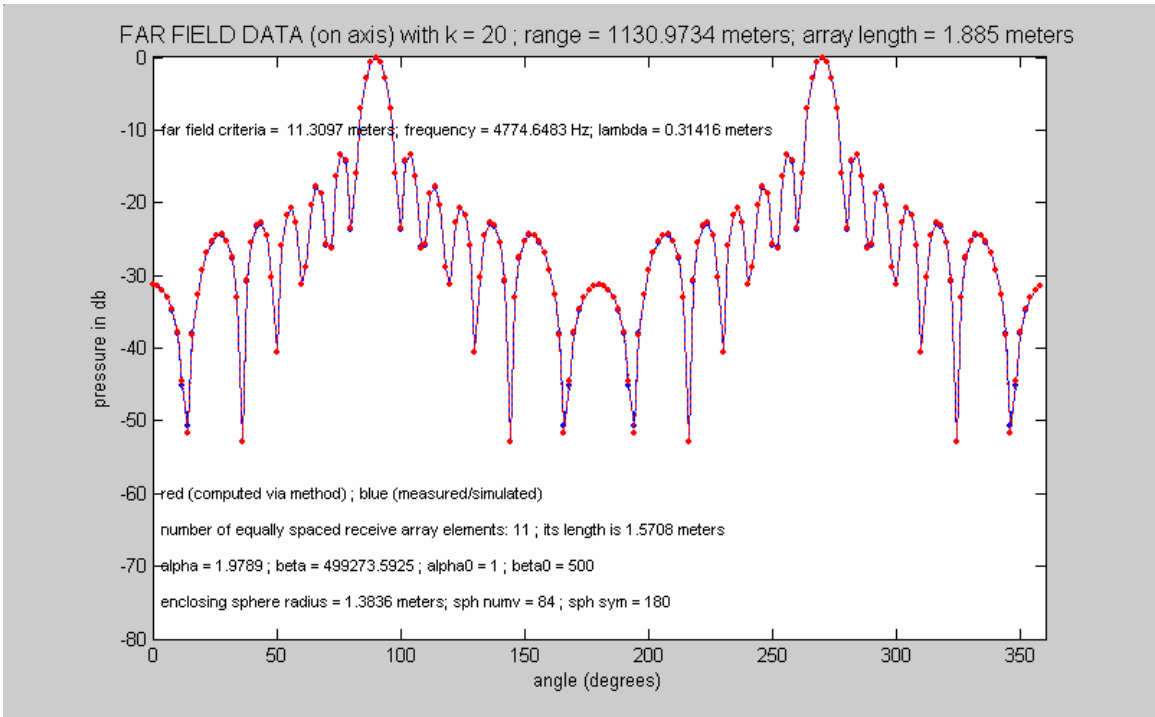


Figure 36 - Far Field Data – computed and simulated
alpha0=1; beta0 = 500

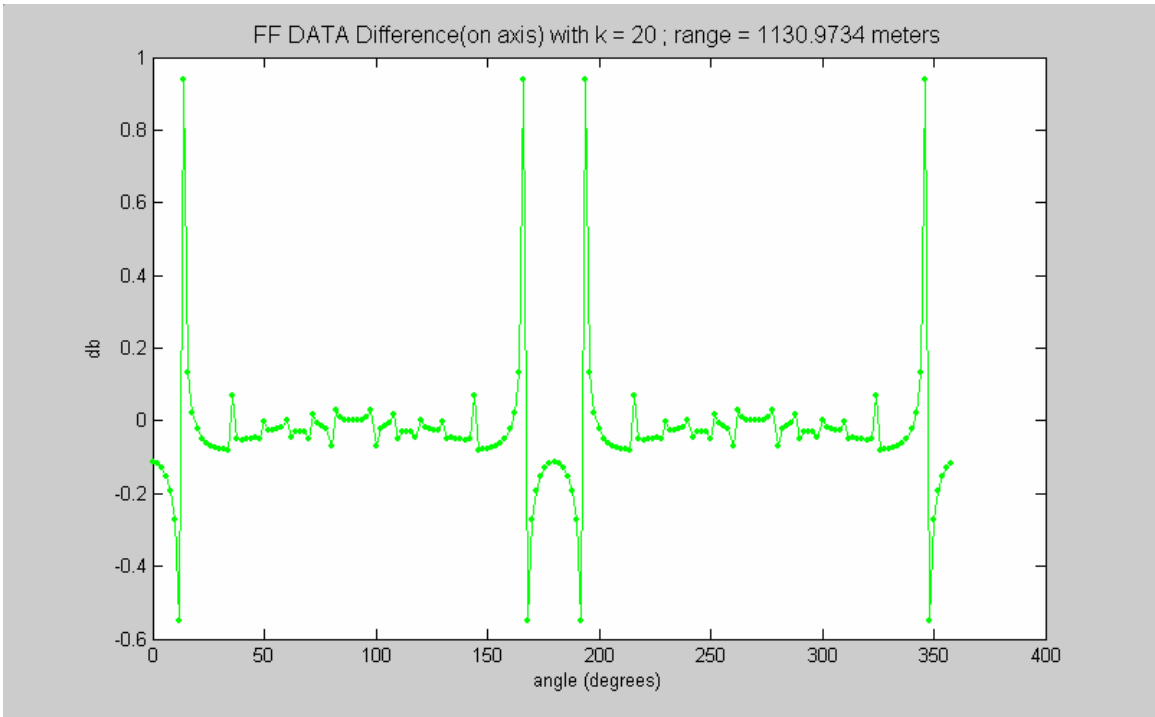


Figure 37 - Far Field Data Difference
alpha0=1; beta0 = 500

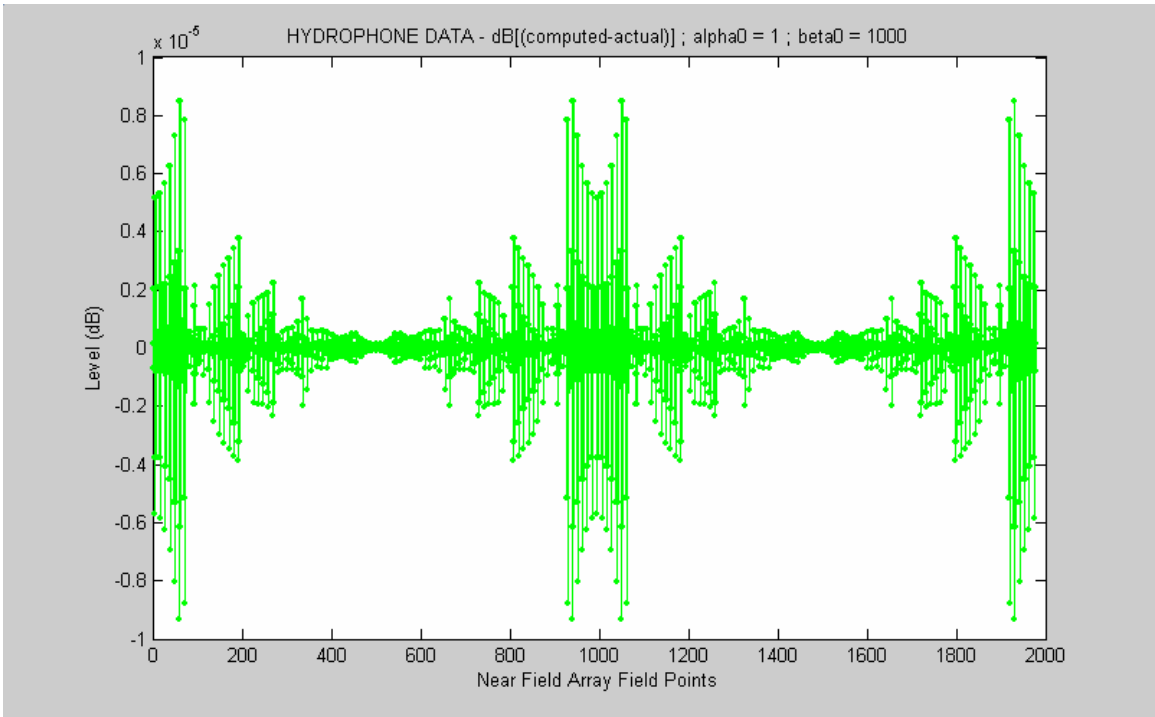


Figure 38 - Near Field Hydrophone Difference Data
alpha0=1; beta0 = 1000

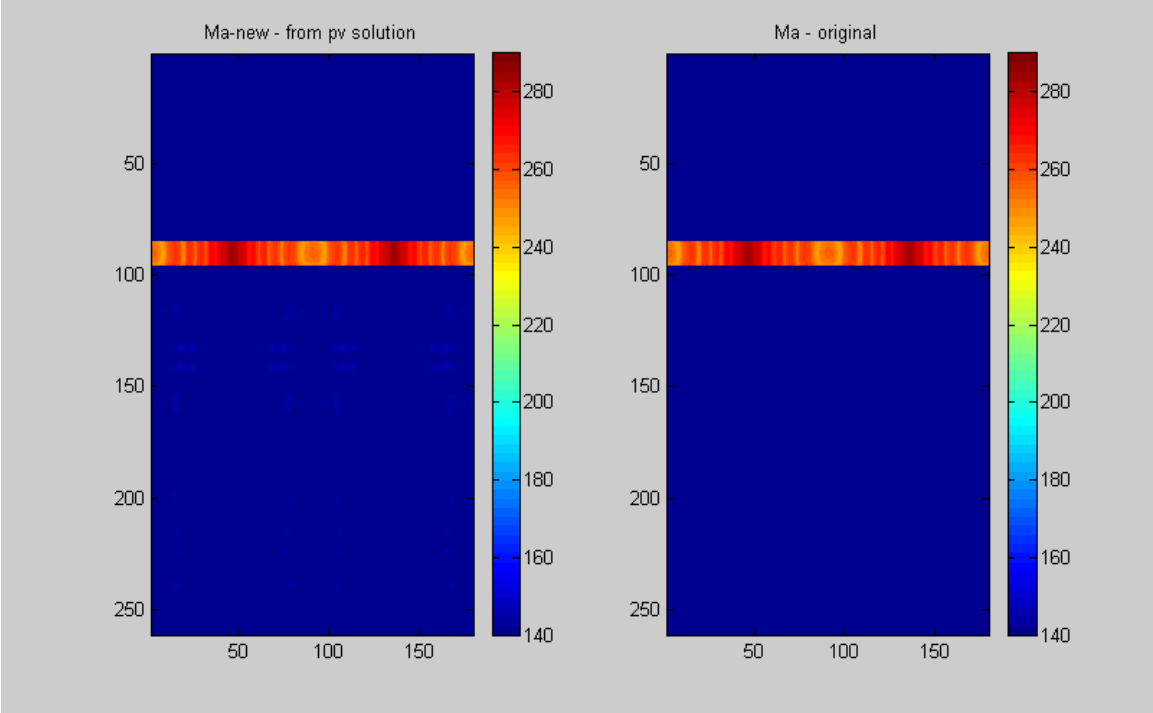


Figure 39 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0 = 1000

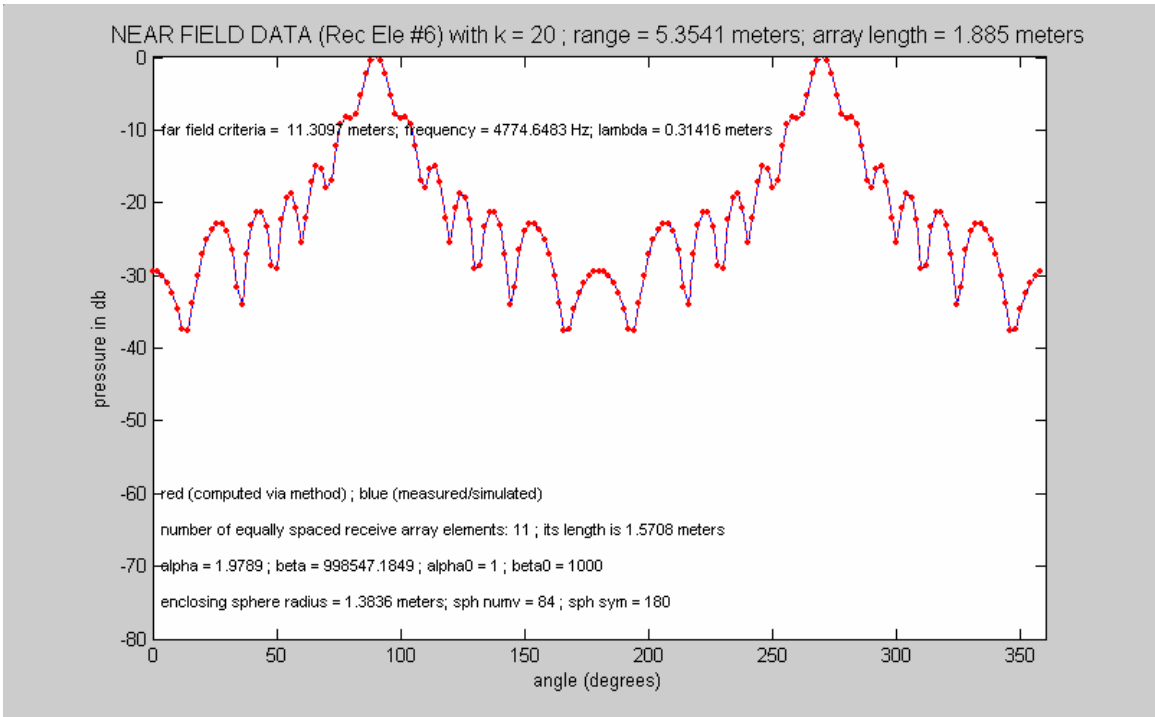


Figure 40 - Near Field Data – computed and simulated
alpha0=1; beta0 = 1000

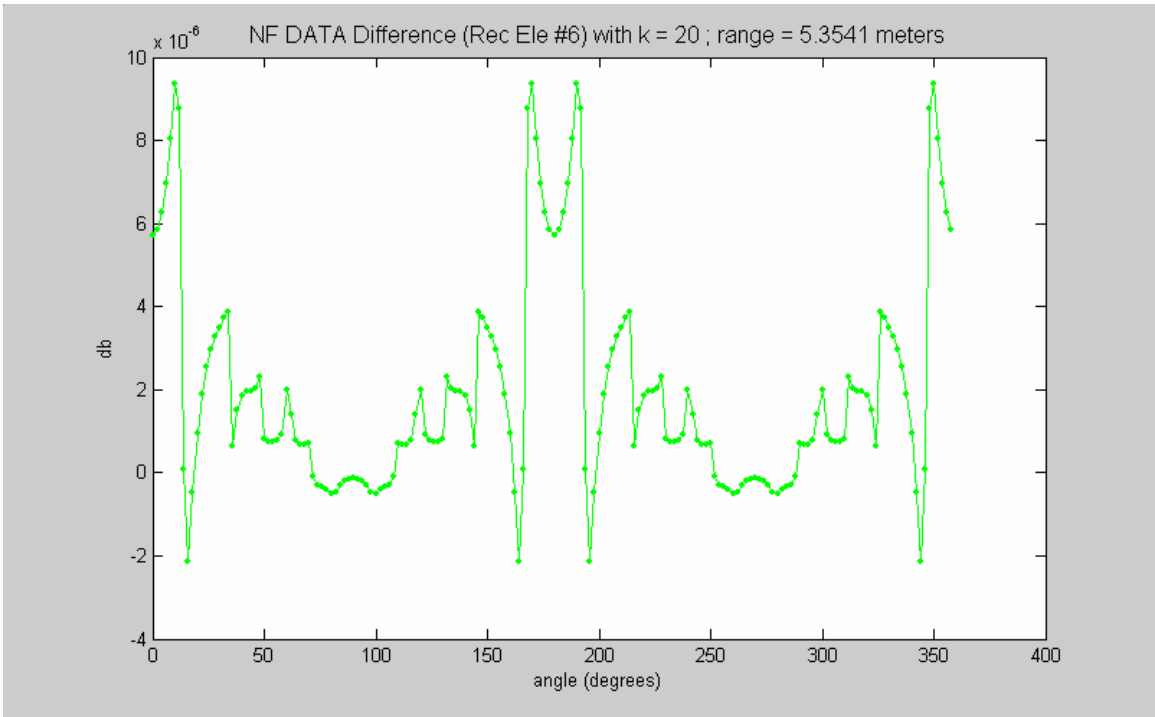


Figure 41 - Near Field Data Difference
alpha0=1; beta0 = 1000

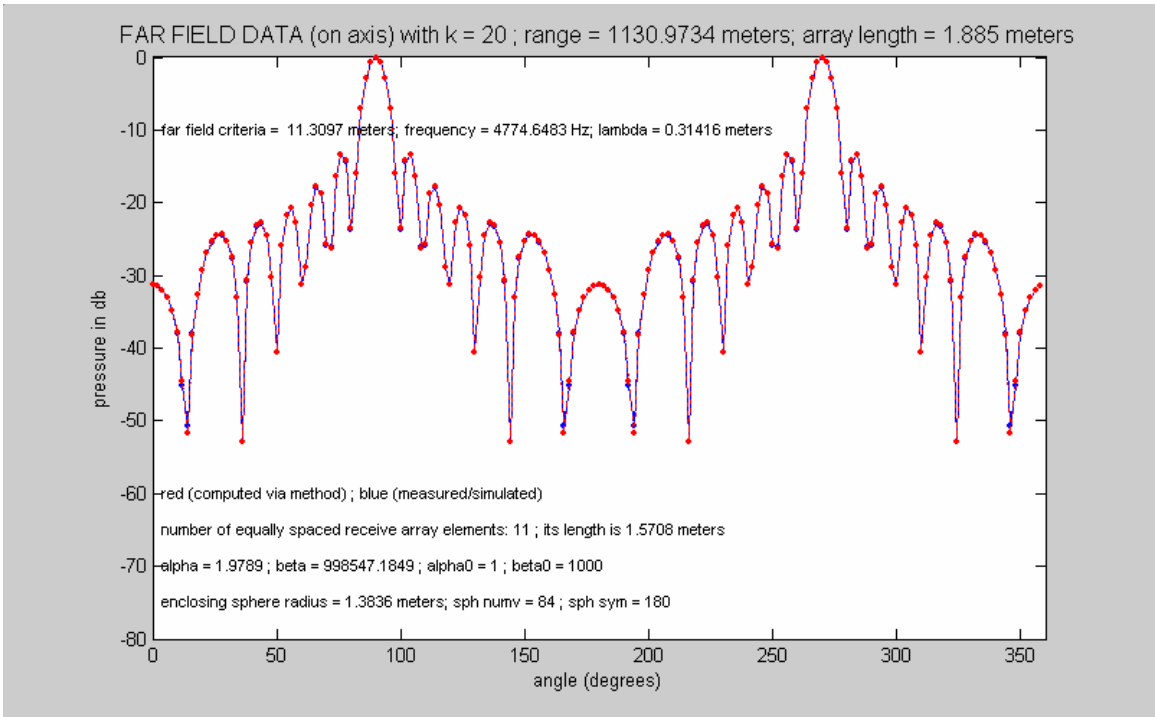


Figure 42 - Far Field Data – computed and simulated
alpha0=1; beta0 = 1000

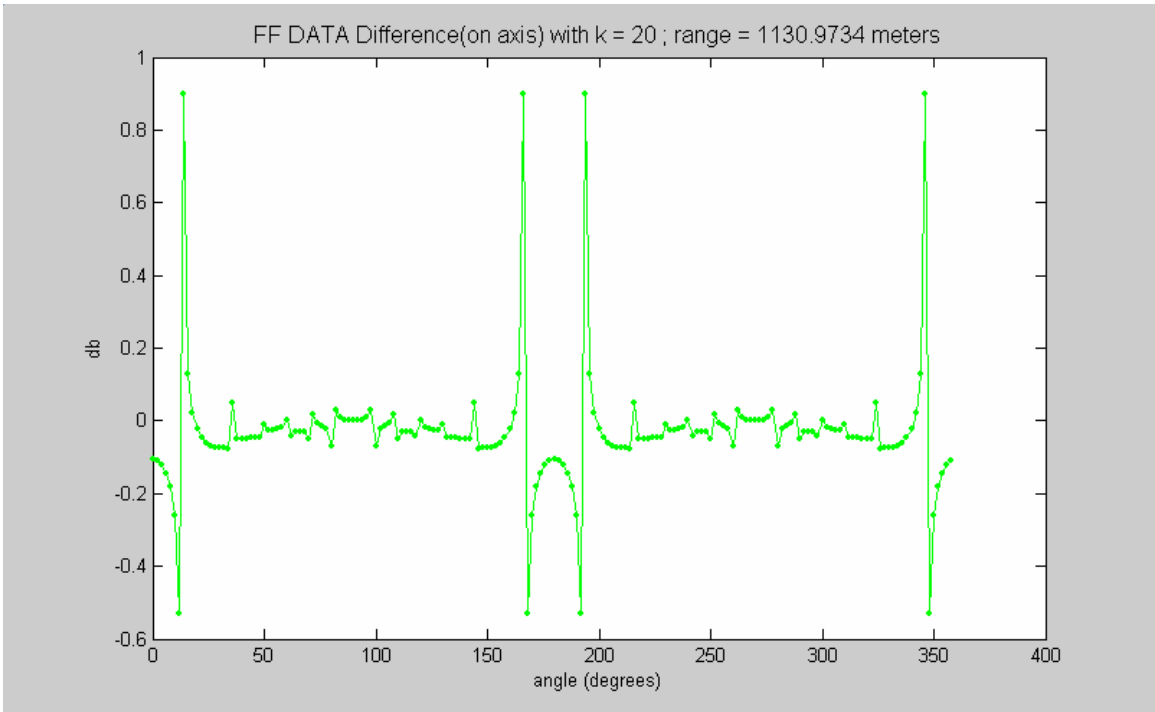


Figure 43 - Far Field Data Difference
alpha0=1; beta0 = 1000

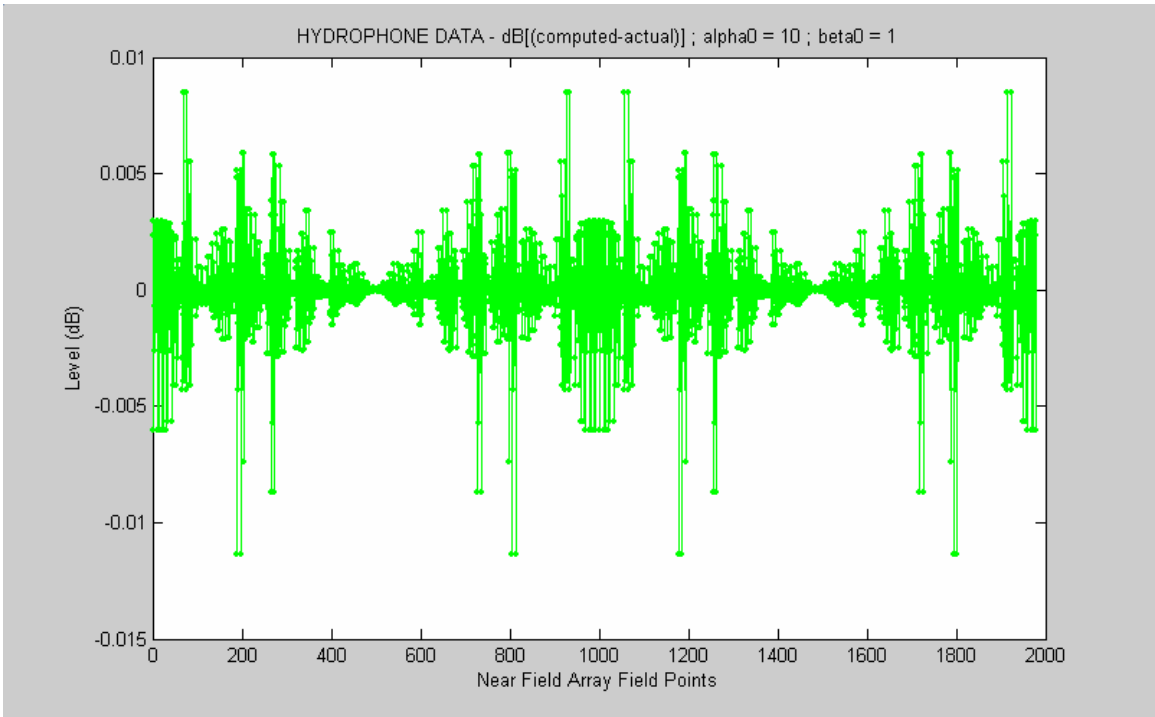


Figure 44 - Near Field Hydrophone Difference Data
alpha0=10; beta0 = 1

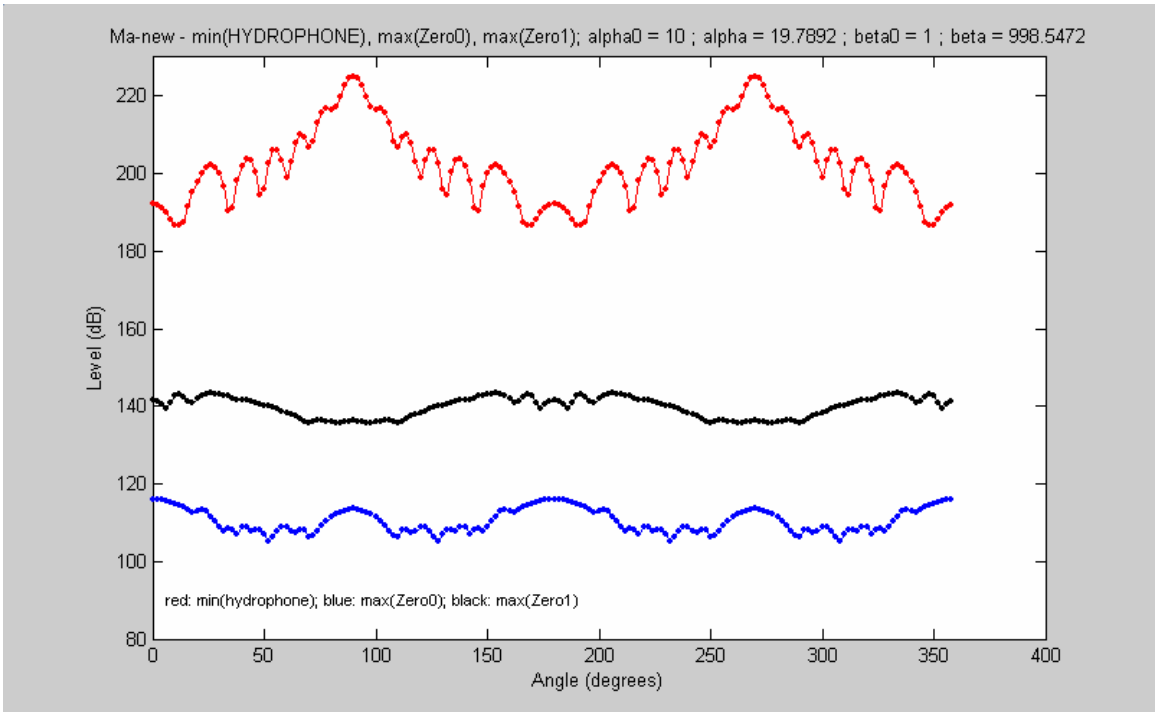


Figure 45 - $\mathbf{M_{a\text{-}new}}$ Components
alpha0=10; beta0 = 1

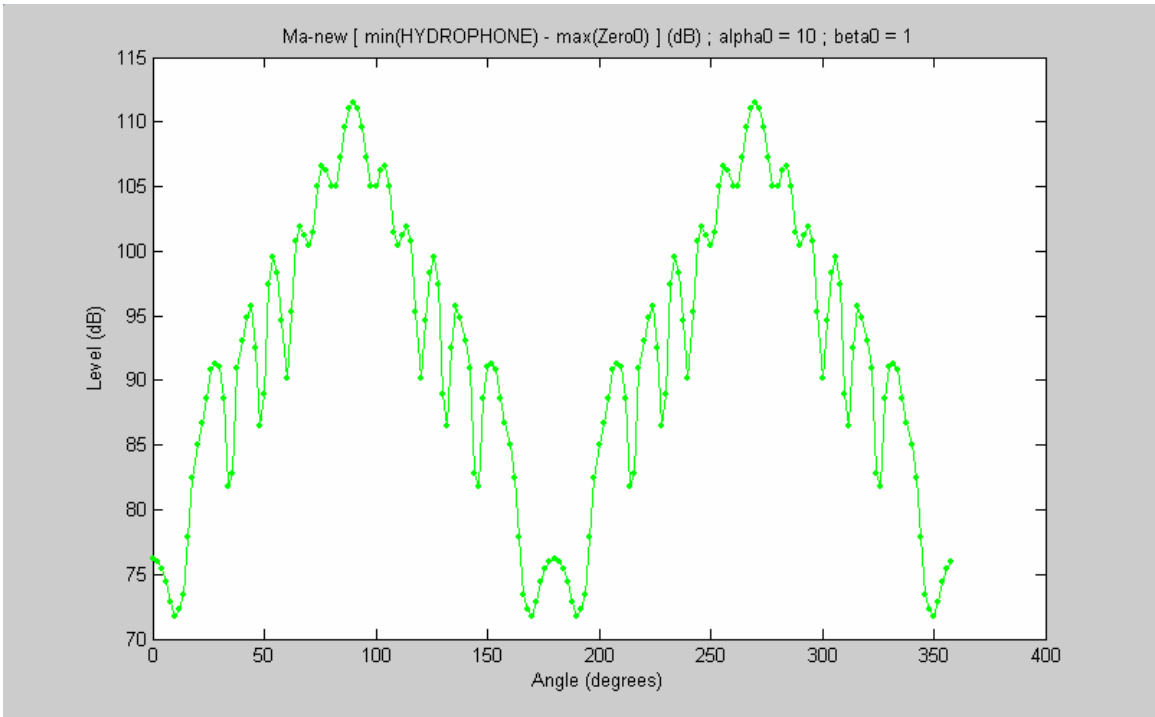


Figure 46 - $\mathbf{M_{a\text{-}new}}$ [min(Hydrophone) – max(Zero0)] Data
alpha0=10; beta0 = 1

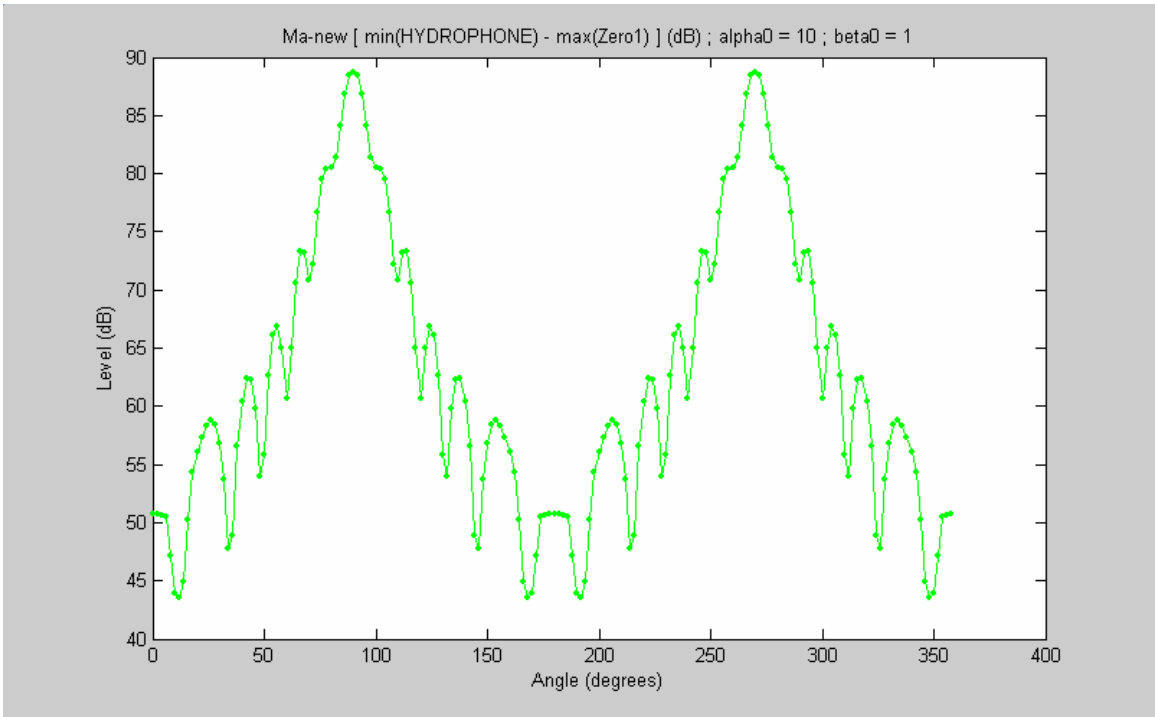


Figure 47 - $\mathbf{M_{a\text{-}new}}$ [min(Hydrophone) – max(Zero1)] Data
alpha0=10; beta0 = 1

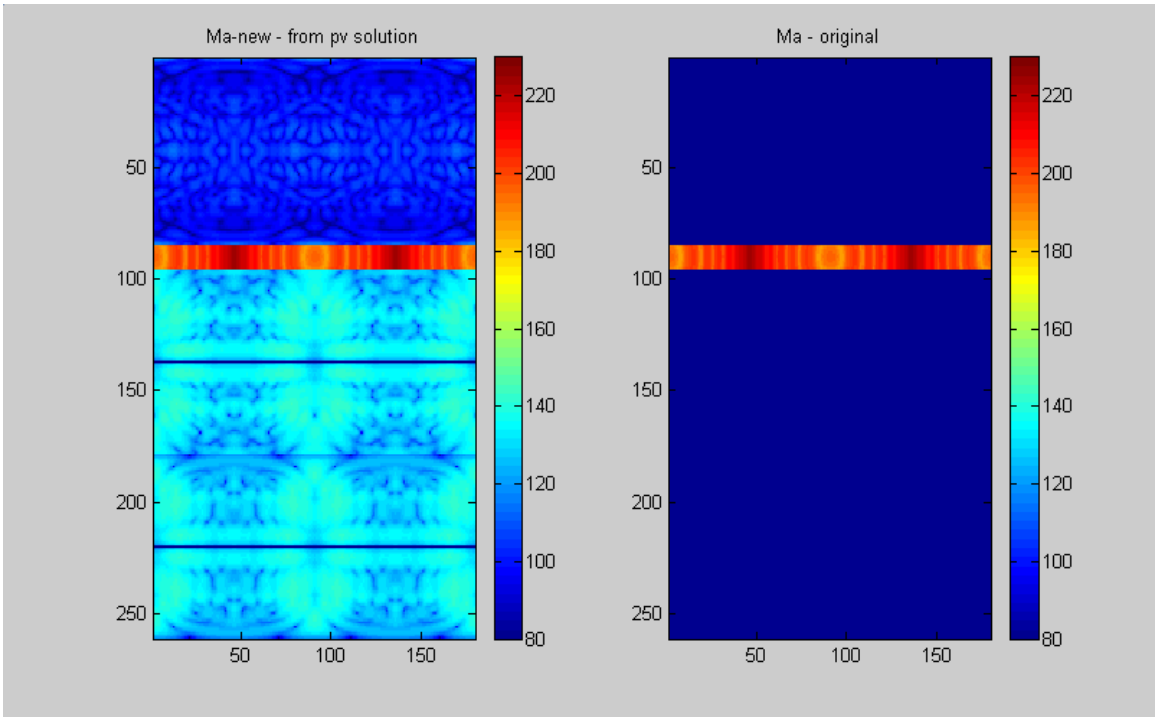


Figure 48 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=10; beta0 = 1

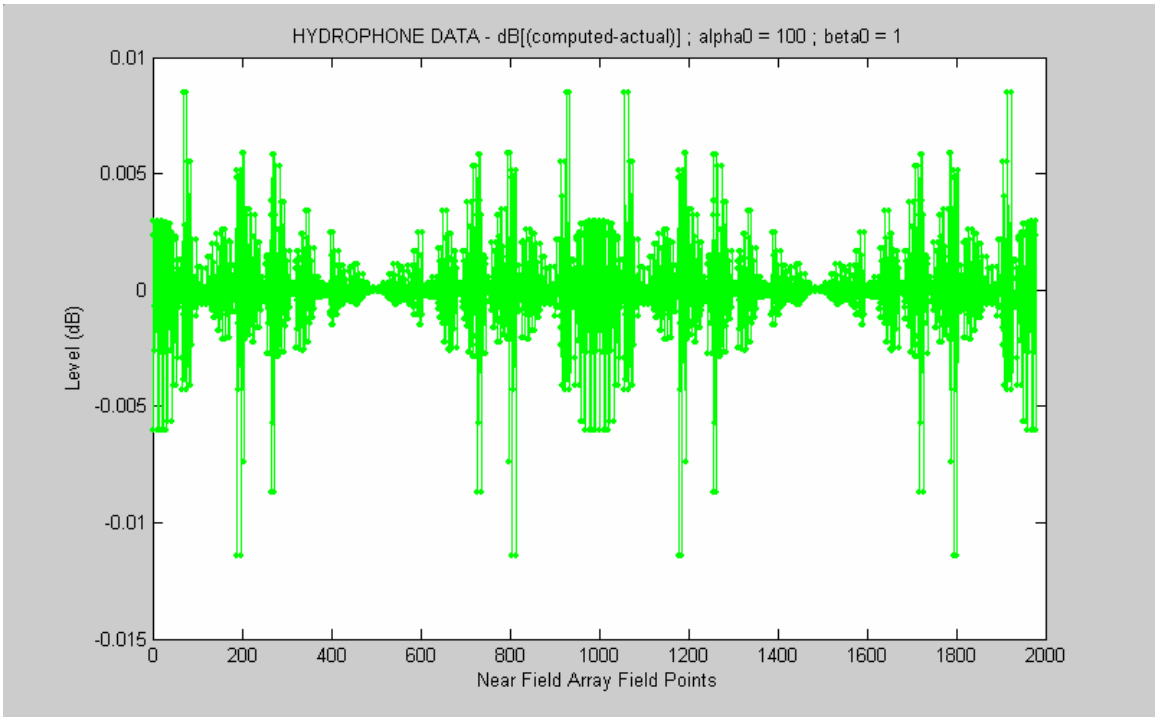


Figure 49 - Near Field Hydrophone Difference Data
alpha0=100; beta0 = 1

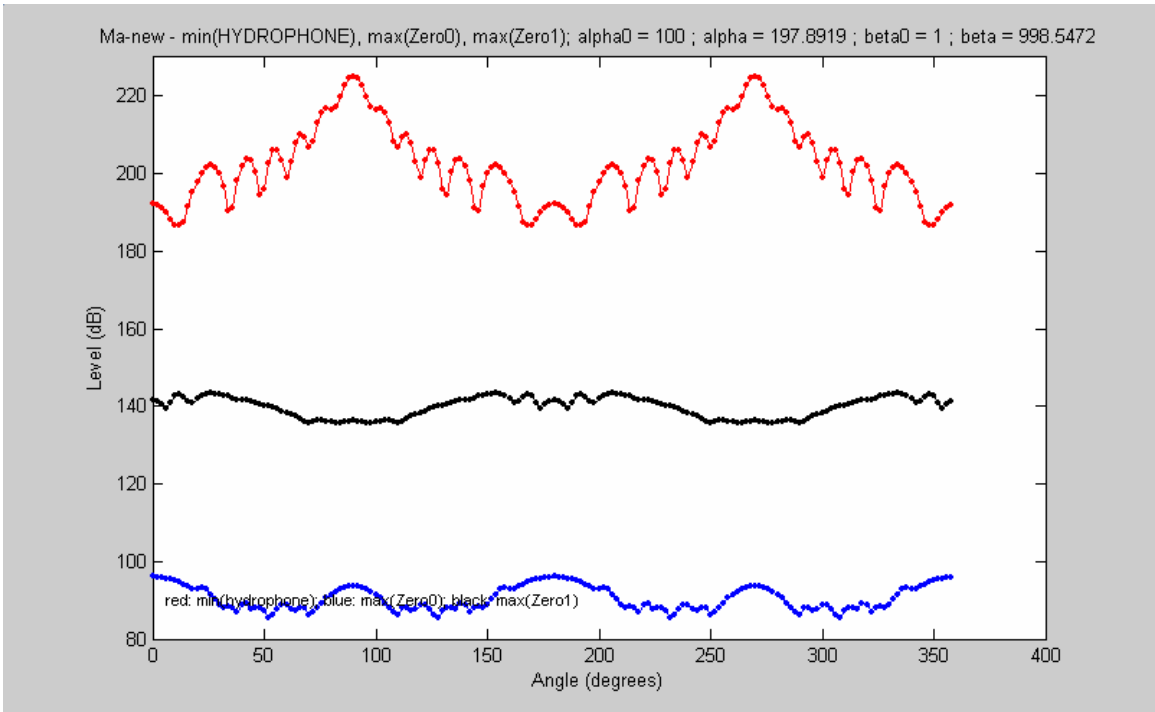


Figure 50 - $\mathbf{M_{a\text{-}new}}$ Components
alpha0=100; beta0 = 1

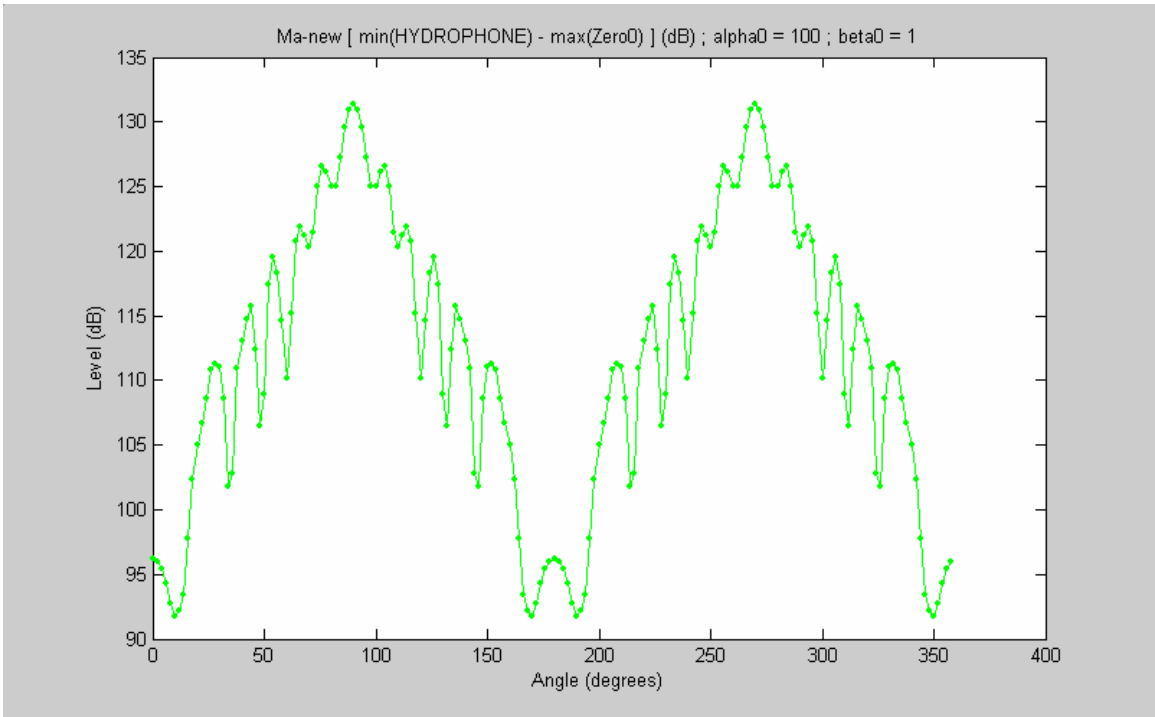


Figure 51 - $\mathbf{M_{a\text{-}new}}$ [min(Hydrophone) – max(Zero0)] Data
alpha0=100; beta0 = 1

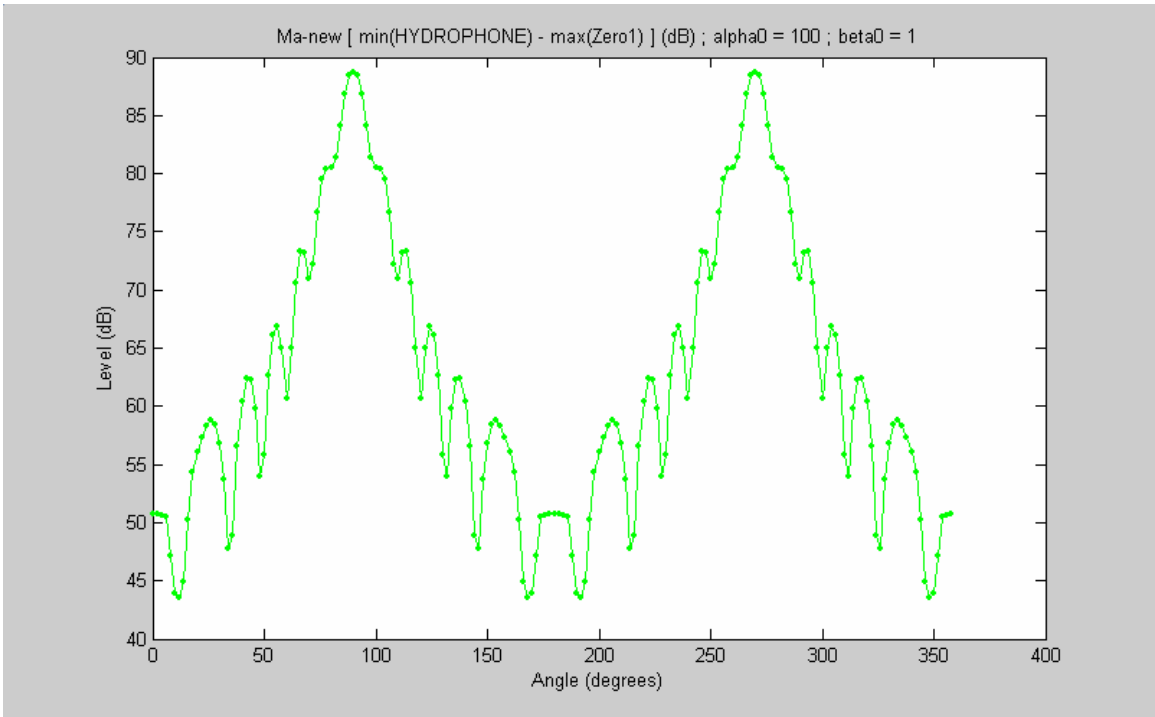


Figure 52 - $\mathbf{M_{a\text{-}new}}$ [min(Hydrophone) – max(Zero1)] Data
alpha0=100; beta0 = 1

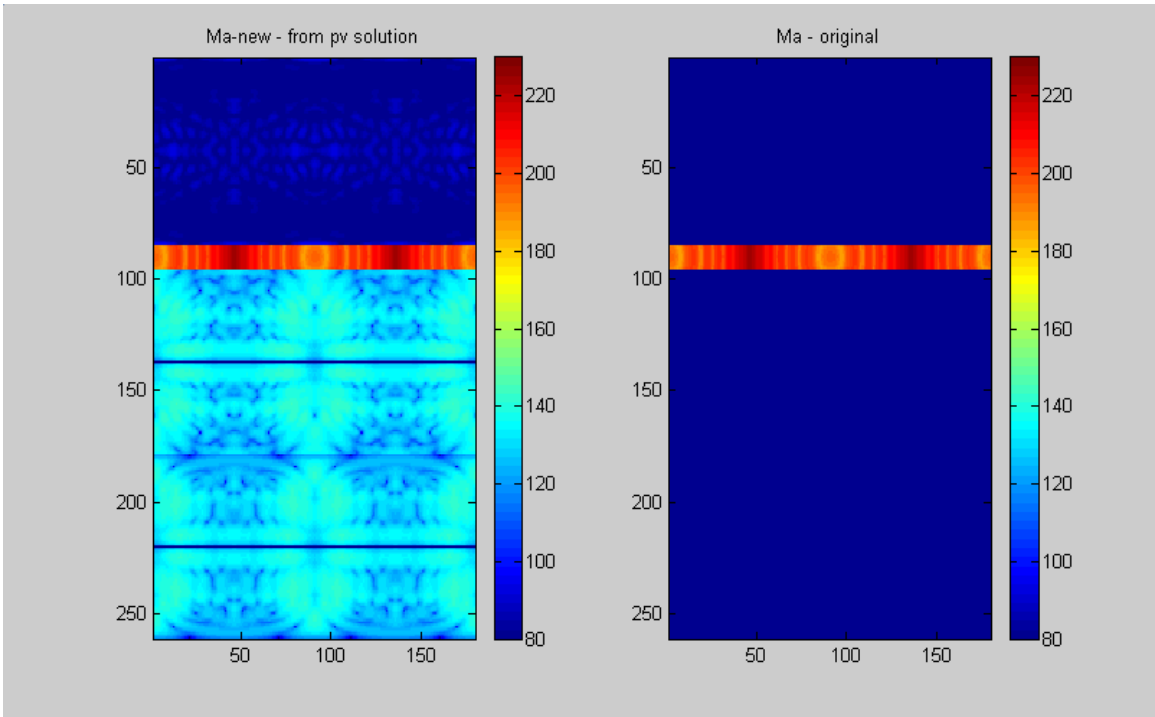


Figure 53 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=100; beta0 = 1

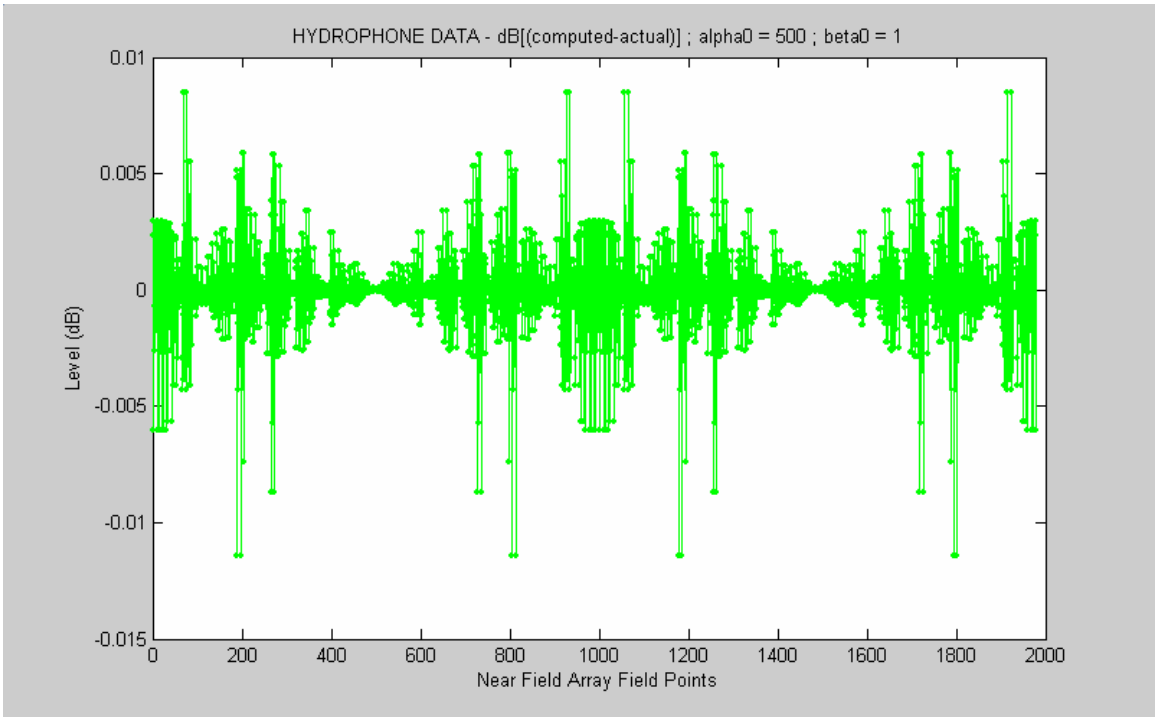


Figure 54 - Near Field Hydrophone Difference Data
alpha0=500; beta0 = 1

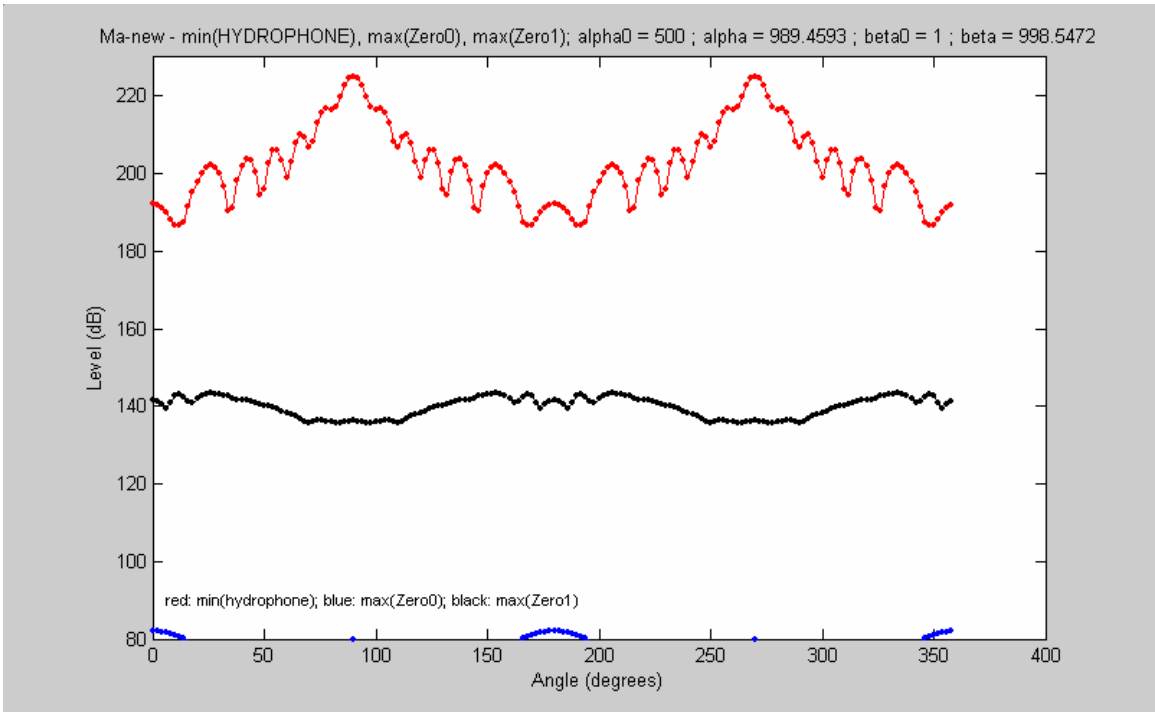


Figure 55 - $\mathbf{M_{a\text{-}new}}$ Components
alpha0=500; beta0 = 1

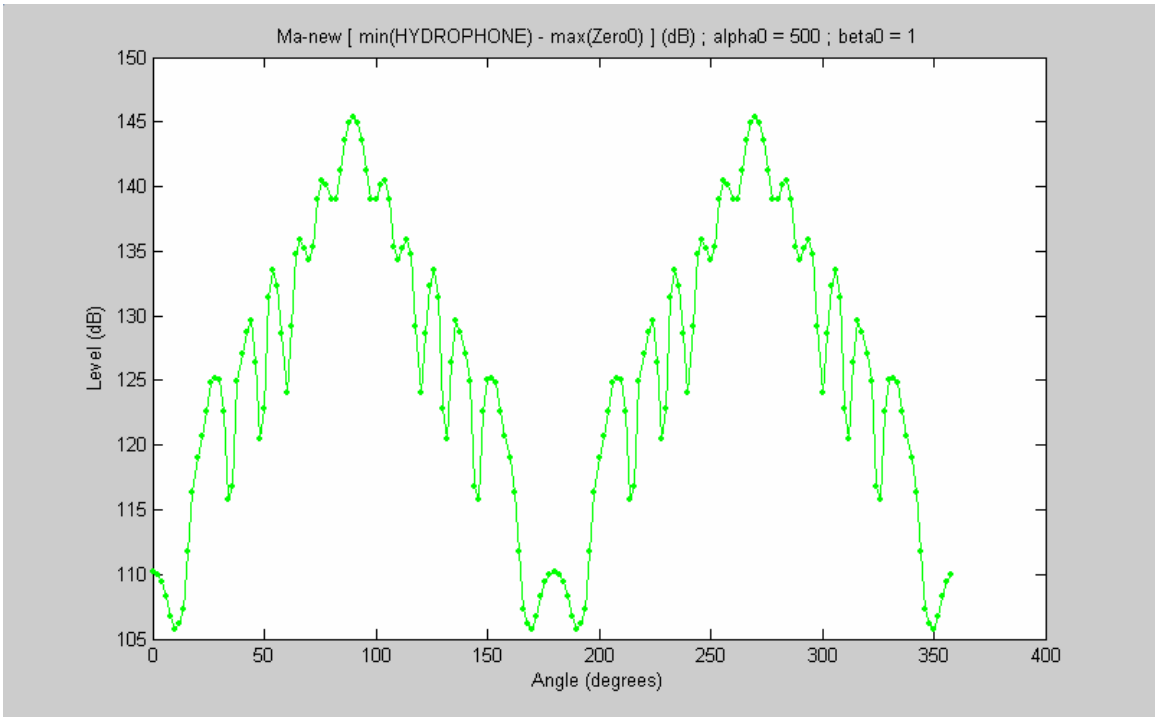


Figure 56 - $\mathbf{M_{a\text{-}new}}$ [min(Hydrophone) – max(Zero0)] Data
alpha0=500; beta0 = 1

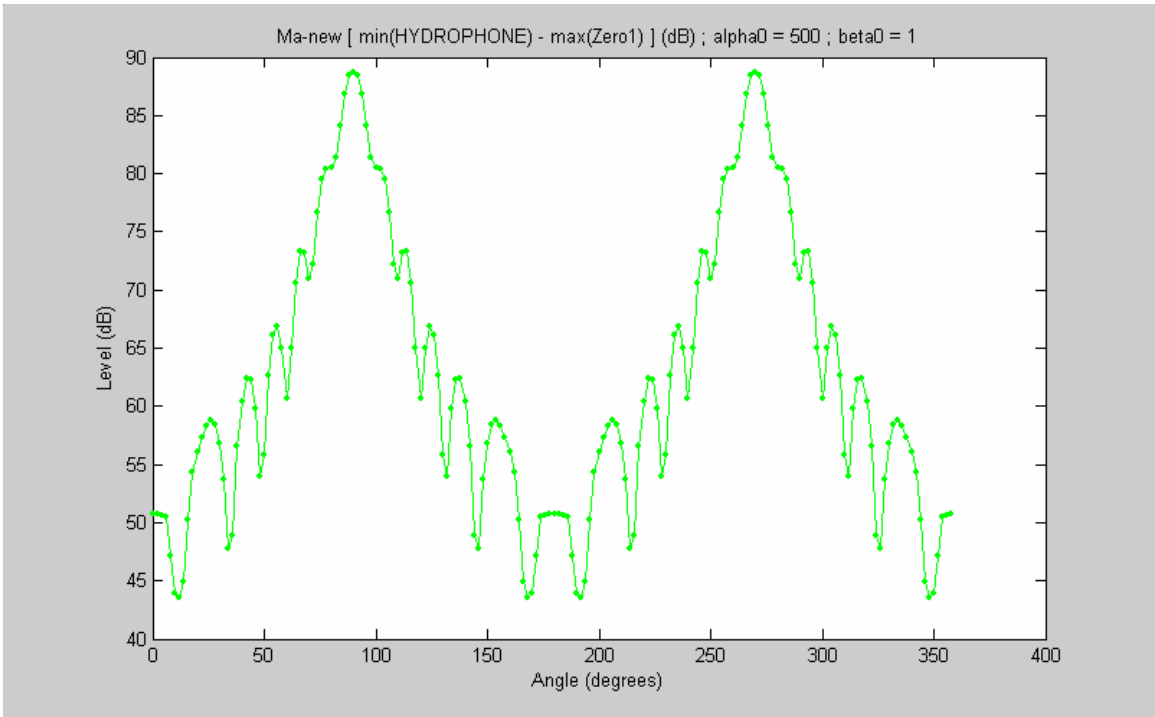


Figure 57 - $\mathbf{M_{a\text{-}new}}$ [min(Hydrophone) – max(Zero1)] Data
alpha0=500; beta0 = 1

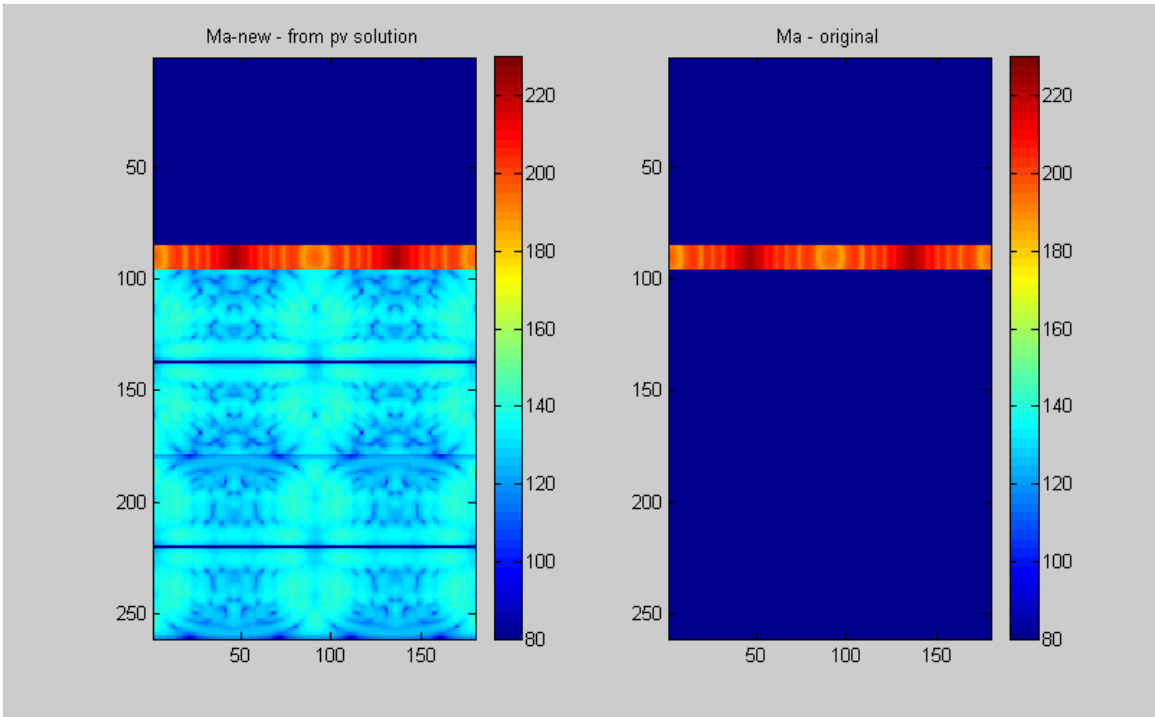


Figure 58 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=500; beta0 = 1

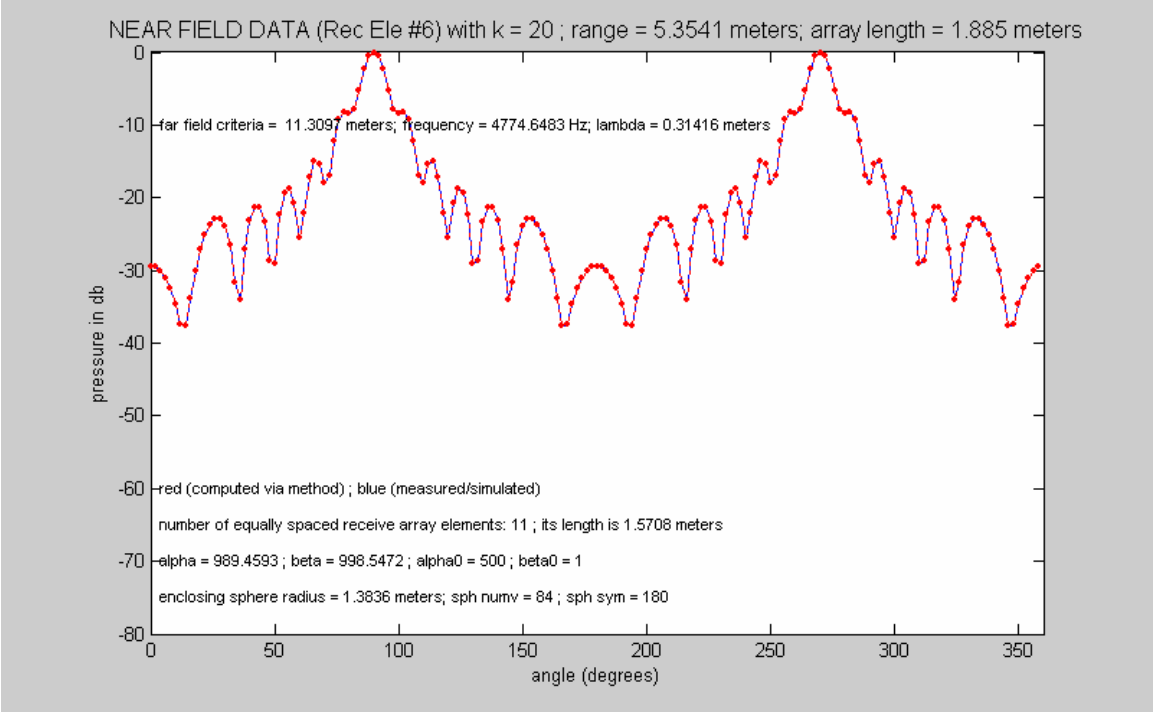


Figure 59 - Near Field Data – computed and simulated
alpha0=500; beta0 = 1

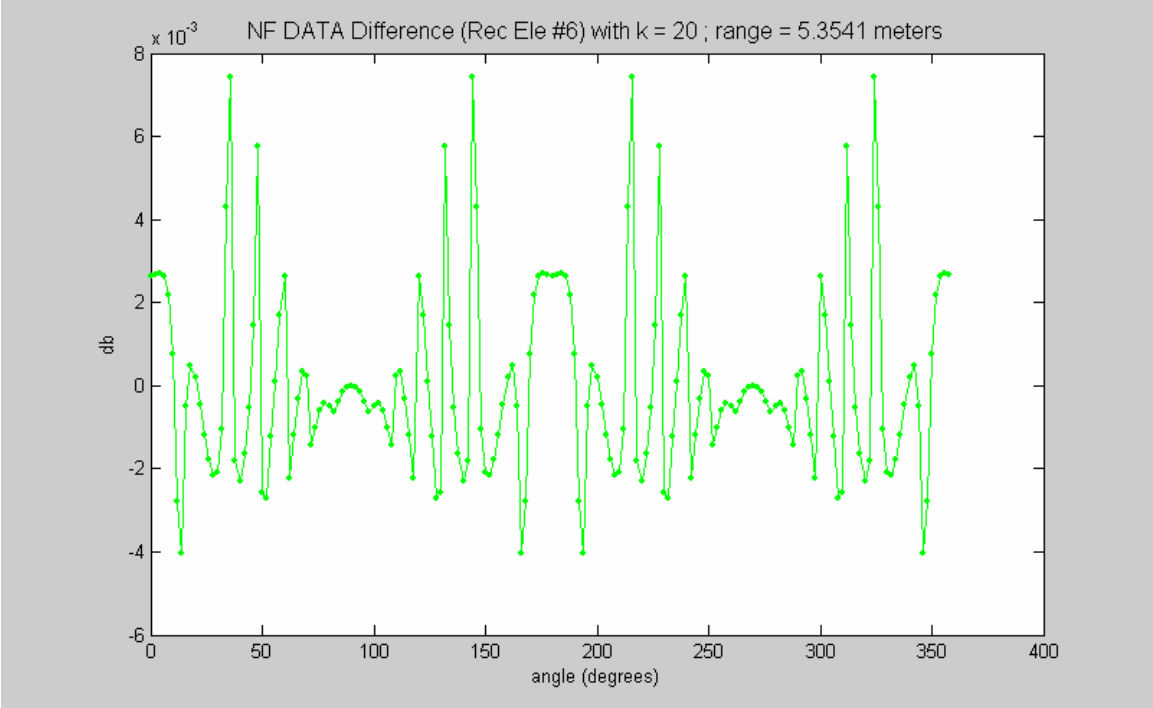


Figure 60 - Near Field Data Difference
alpha0=500; beta0 = 1

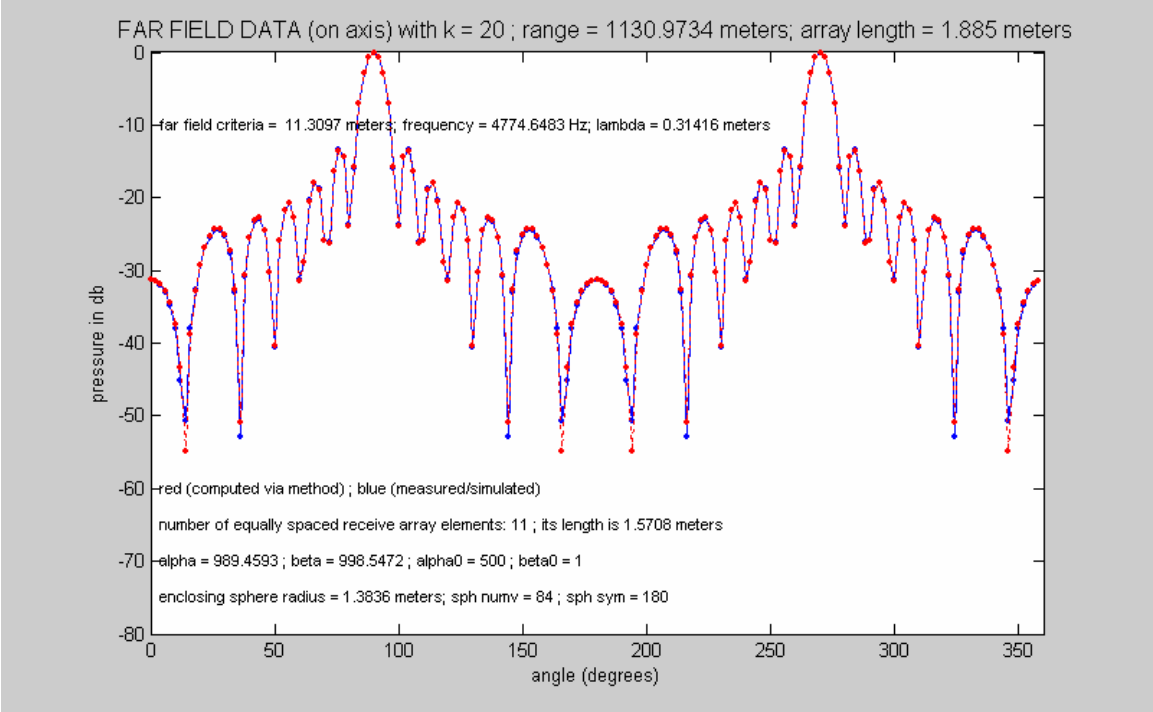


Figure 61 - Far Field Data – computed and simulated
alpha0=500; beta0 = 1

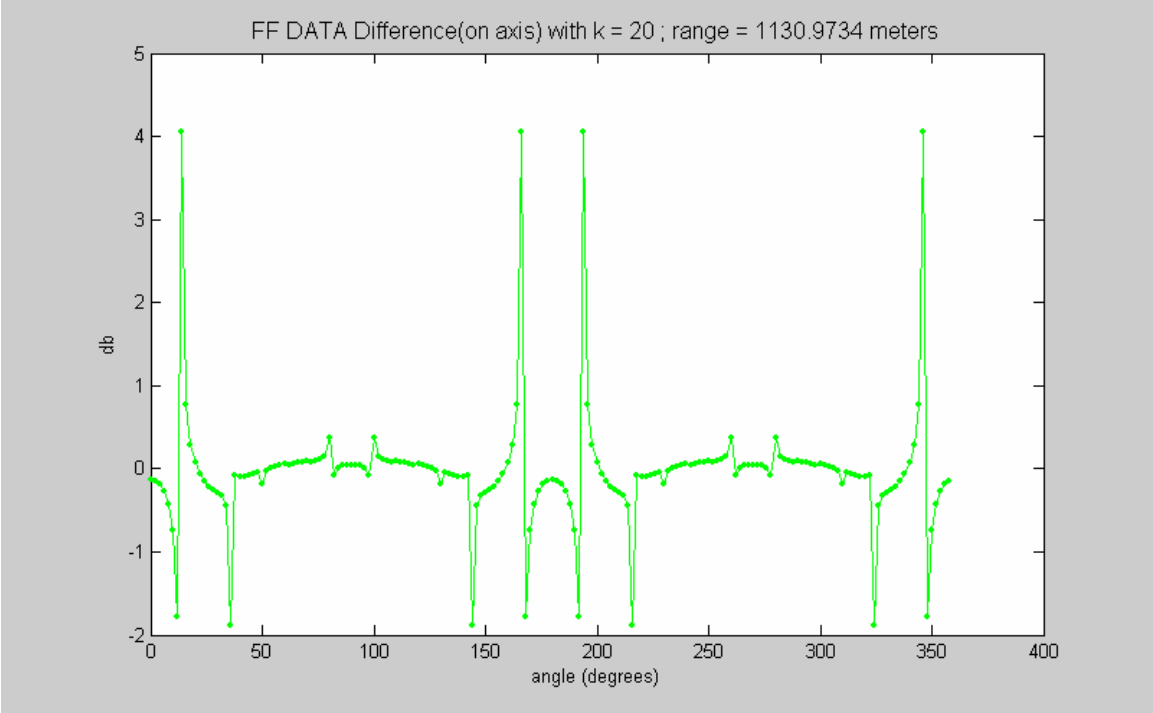


Figure 62 - Far Field Data Difference
alpha0=500; beta0 = 1

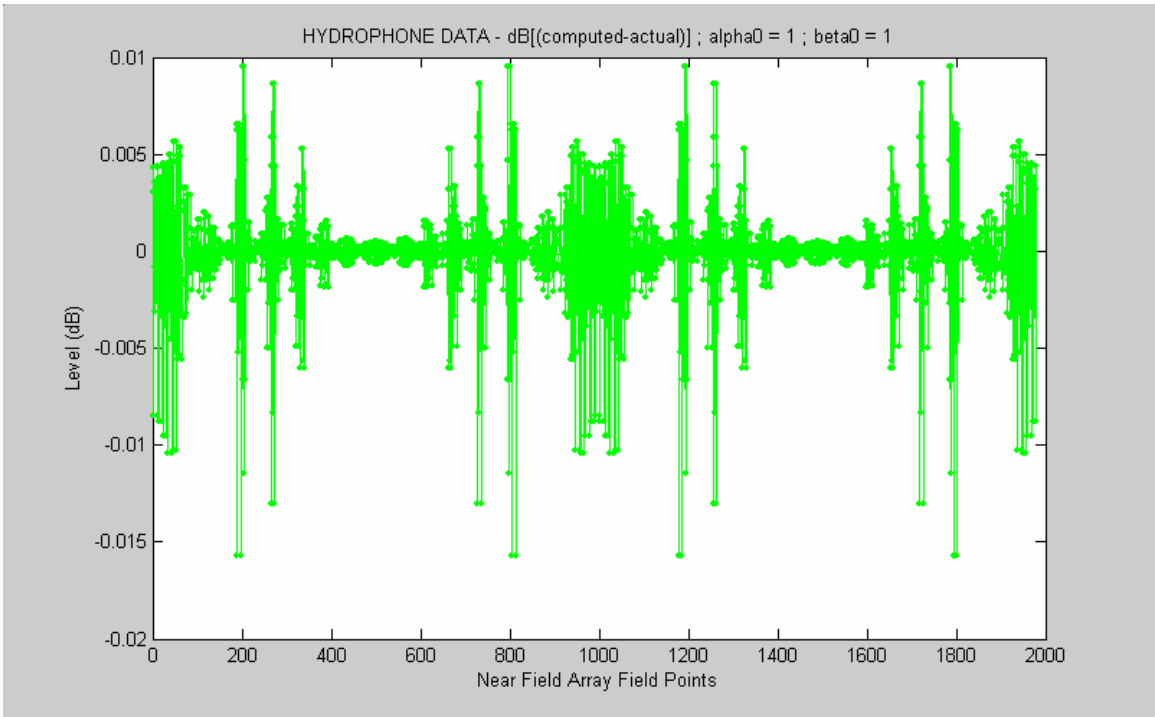


Figure 63 - Near Field Hydrophone Difference Data
alpha0=1; beta0=1; range error = -2 *lambda

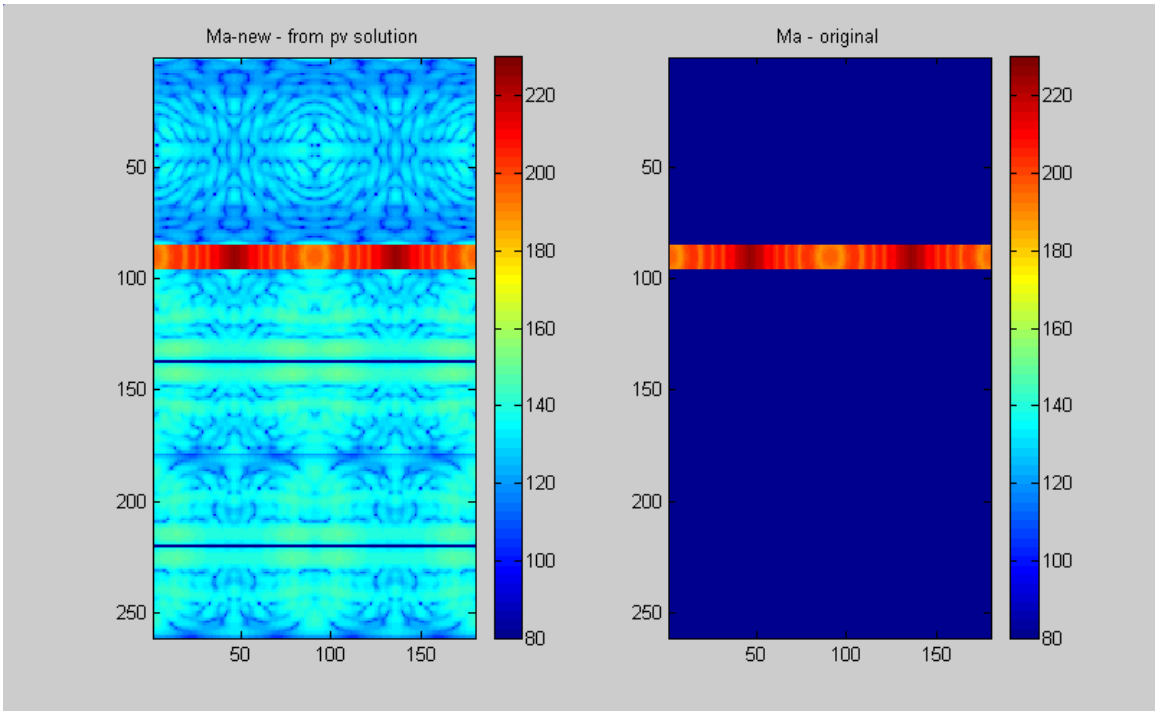


Figure 64 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0=1; range error = -2 *lambda

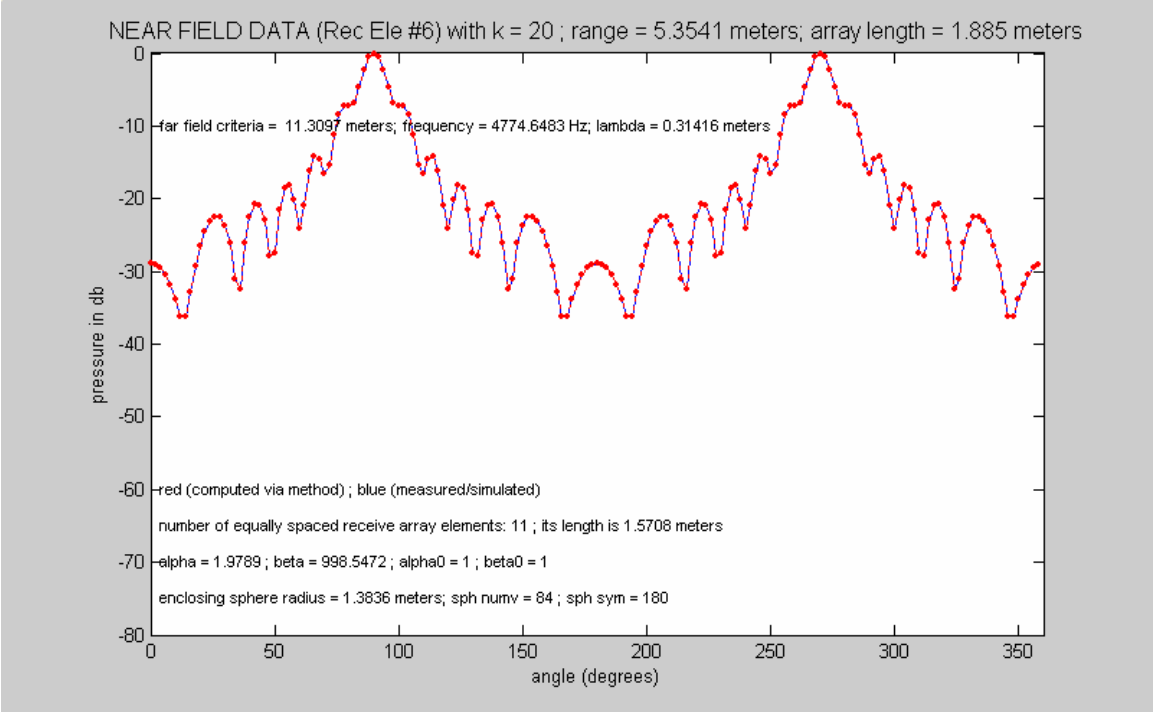


Figure 65 - Near Field Data – computed and simulated
alpha0=1; beta0=1; range error = -2 *lambda

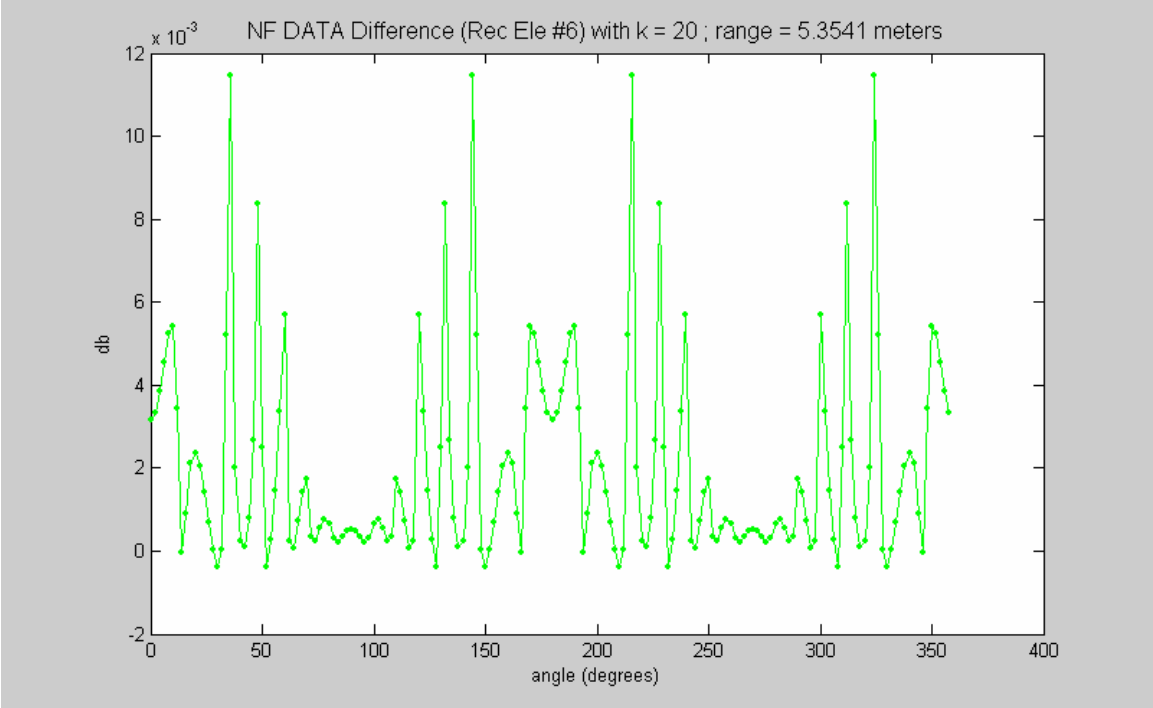


Figure 66 - Near Field Data Difference
alpha0=1; beta0=1; range error = -2 *lambda

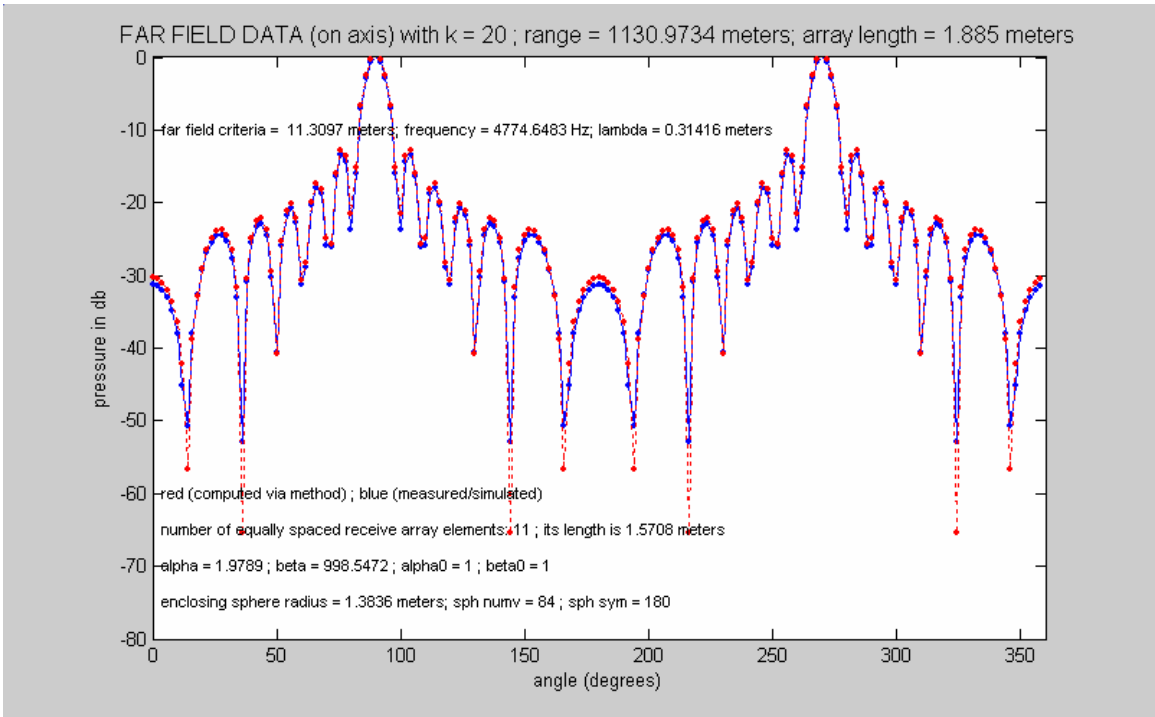


Figure 67 - Far Field Data – computed and simulated
alpha0=1; beta0=1; range error = -2 *lambda

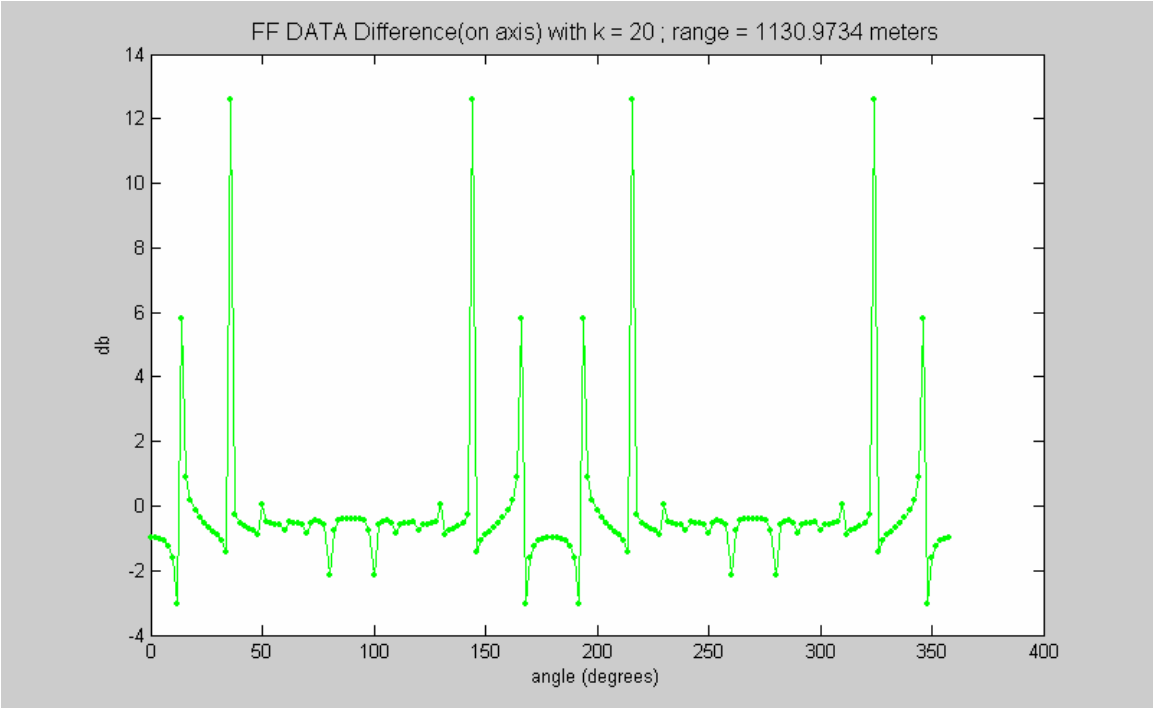


Figure 68 - Far Field Data Difference
alpha0=1; beta0=1; range error = -2 *lambda

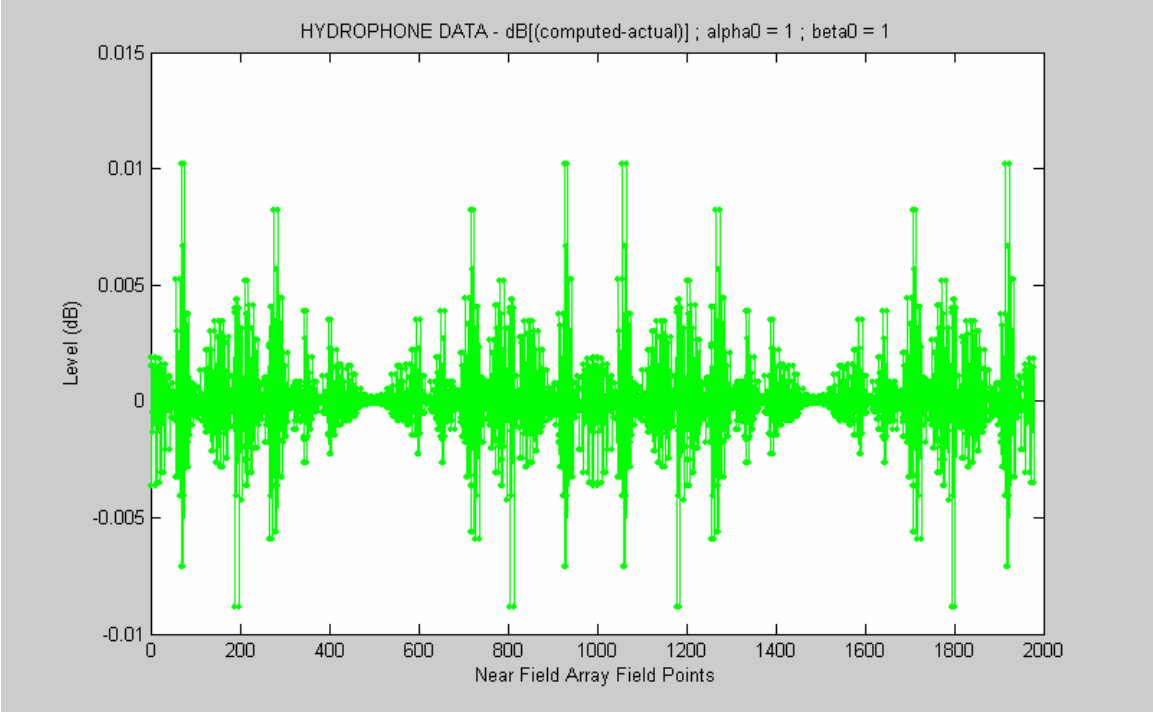


Figure 69 - Near Field Hydrophone Difference Data
alpha0=1; beta0=1; range error = +2 *lambda

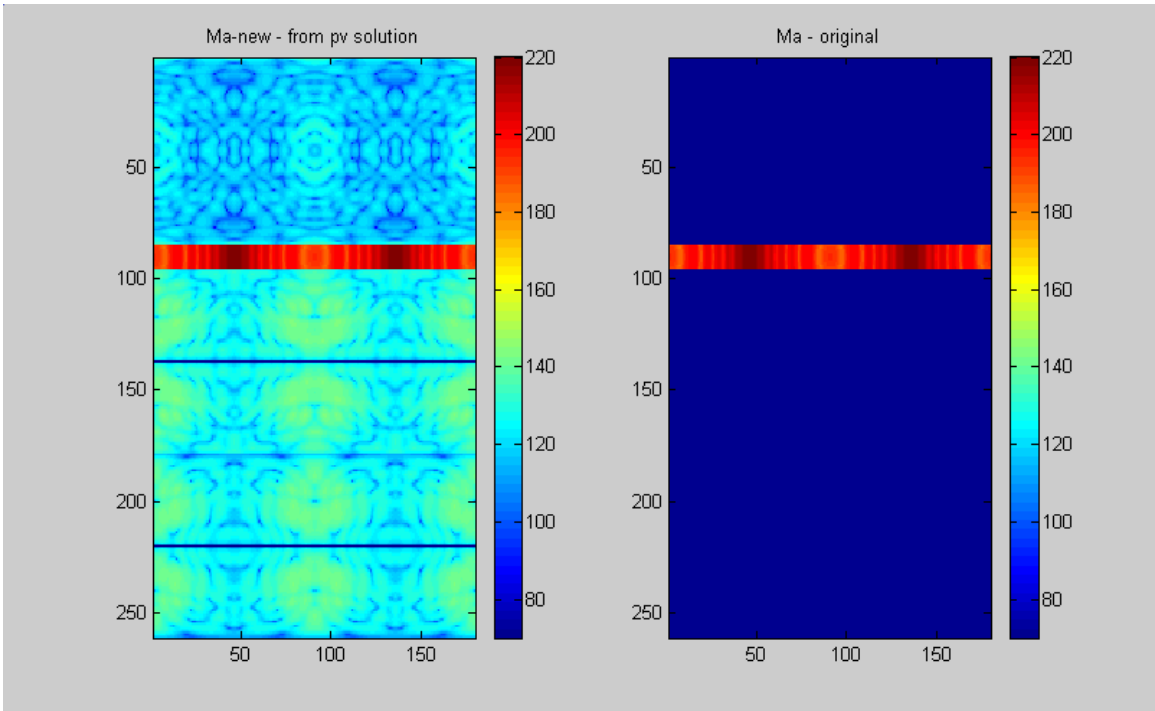


Figure 70 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0=1; range error = +2 *lambda

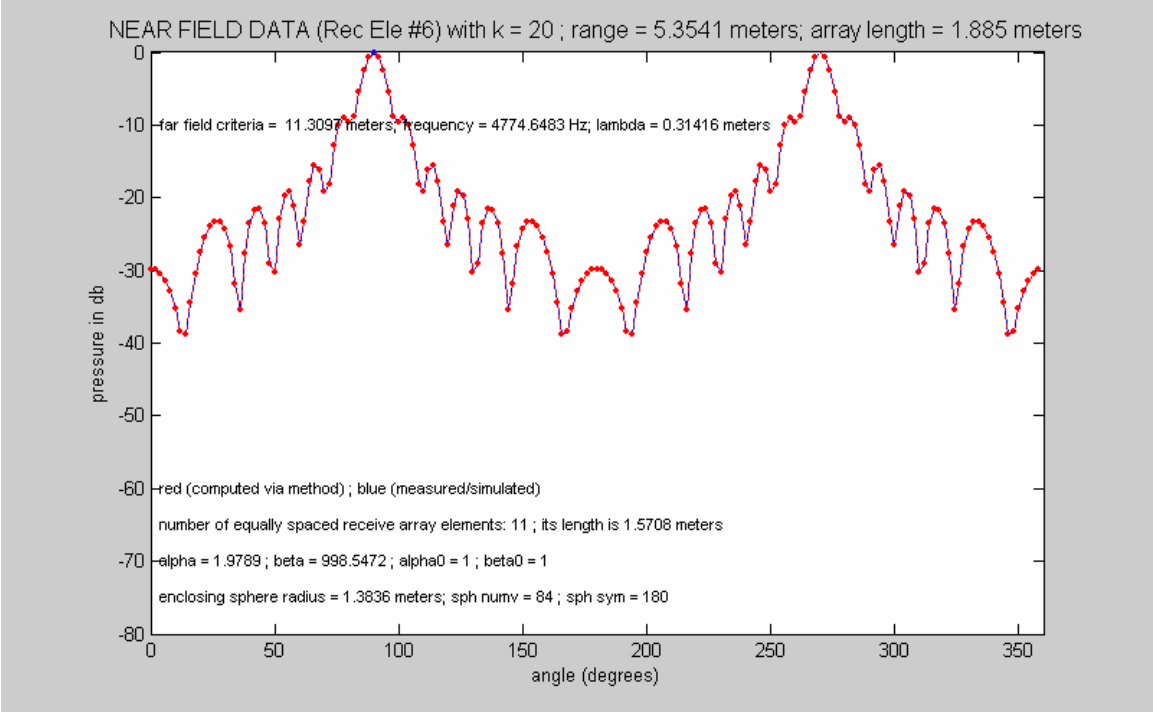


Figure 71 - Near Field Data – computed and simulated
alpha0=1; beta0=1; range error = +2 *lambda

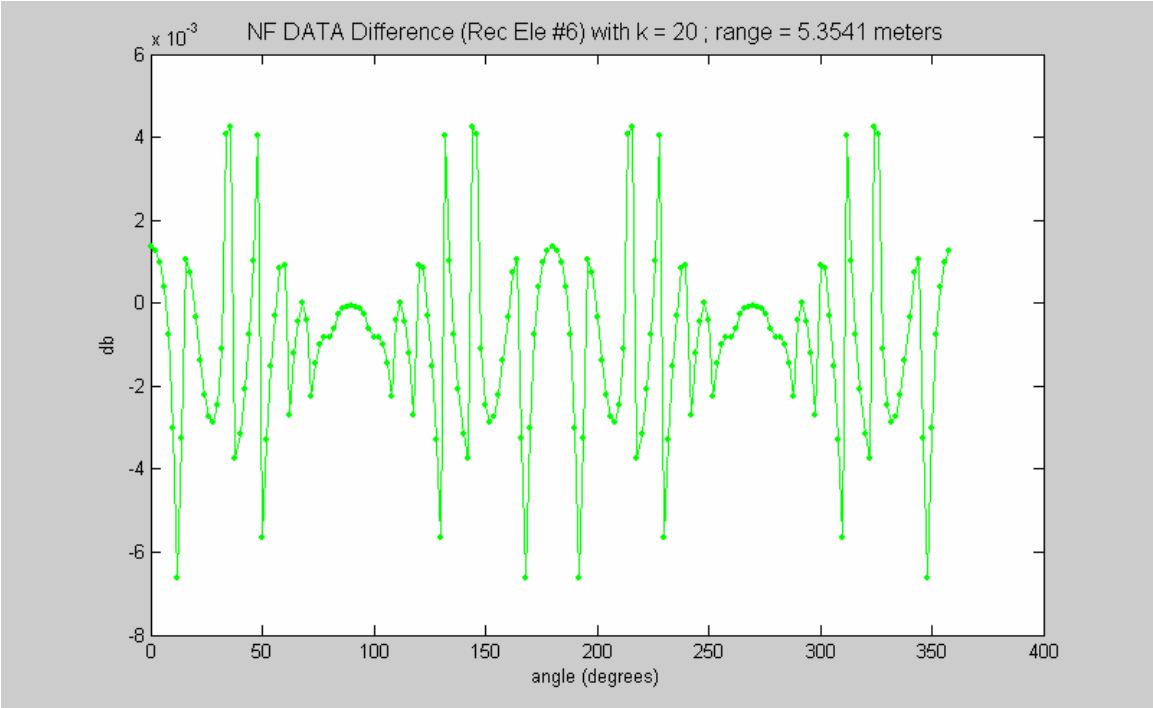


Figure 72 - Near Field Data Difference
alpha0=1; beta0=1; range error = +2 *lambda

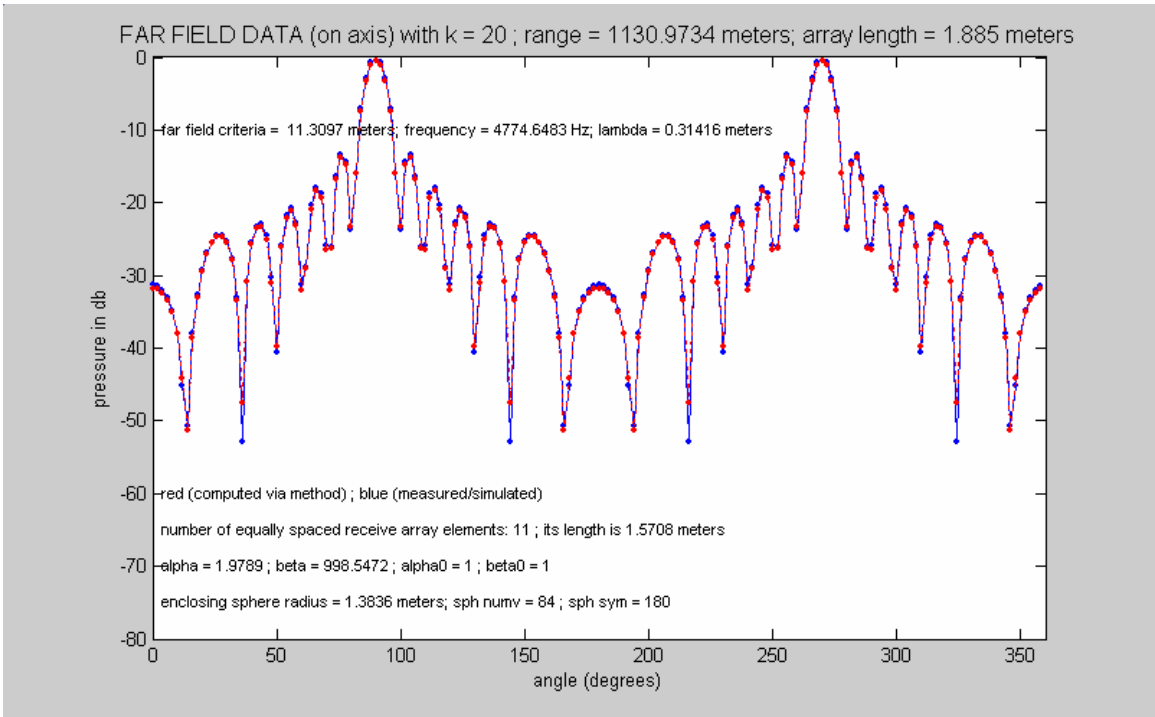


Figure 73 - Far Field Data – computed and simulated
alpha0=1; beta0=1; range error = +2 *lambda

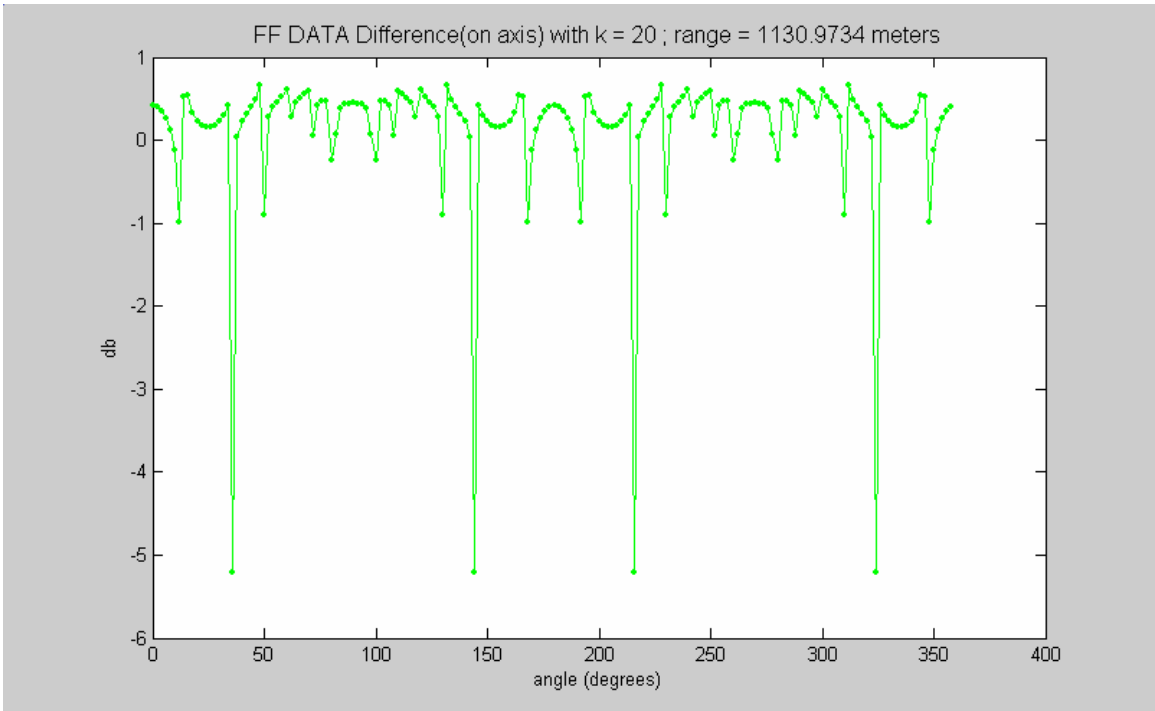


Figure 74 - Far Field Data Difference
alpha0=1; beta0=1; range error = +2 *lambda

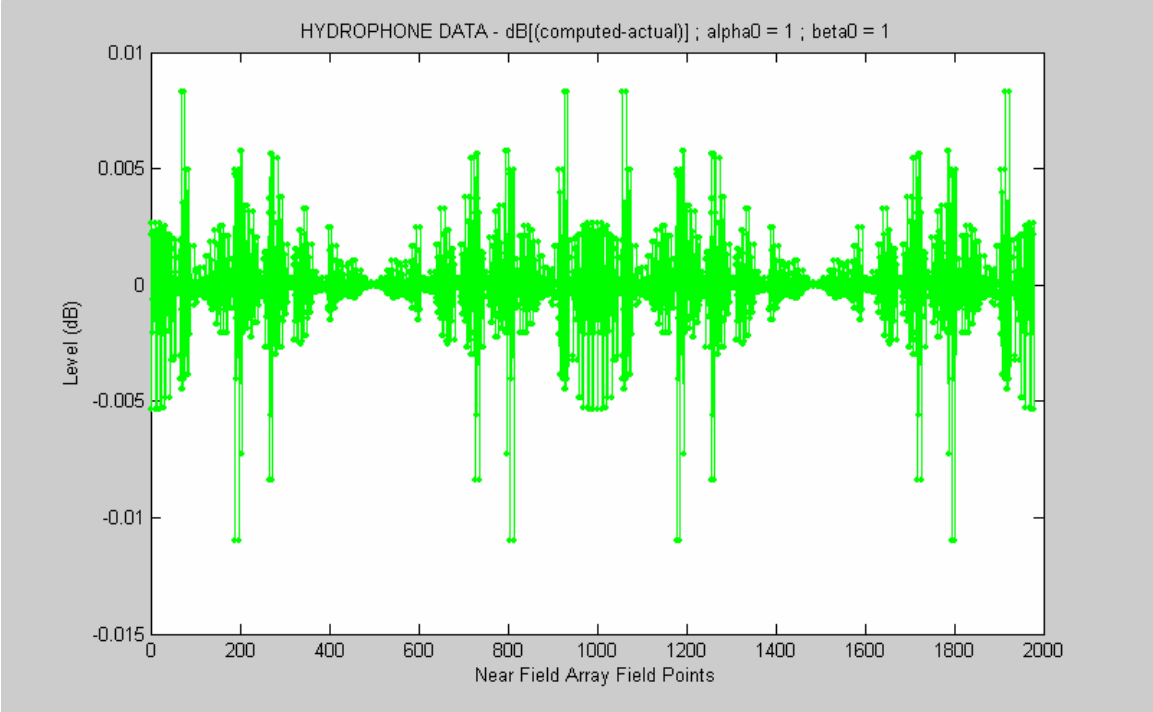


Figure 75 - Near Field Hydrophone Difference Data
alpha0=1; beta0=1; range error = -1 cm

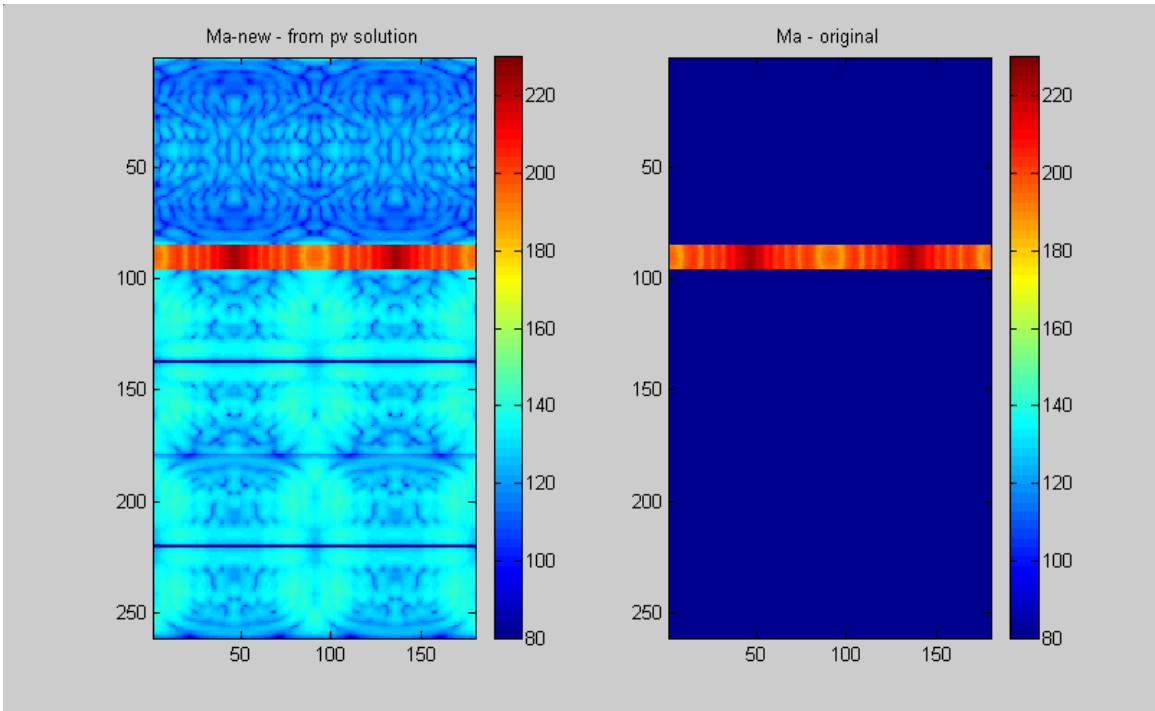


Figure 76 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0=1; range error = -1 cm

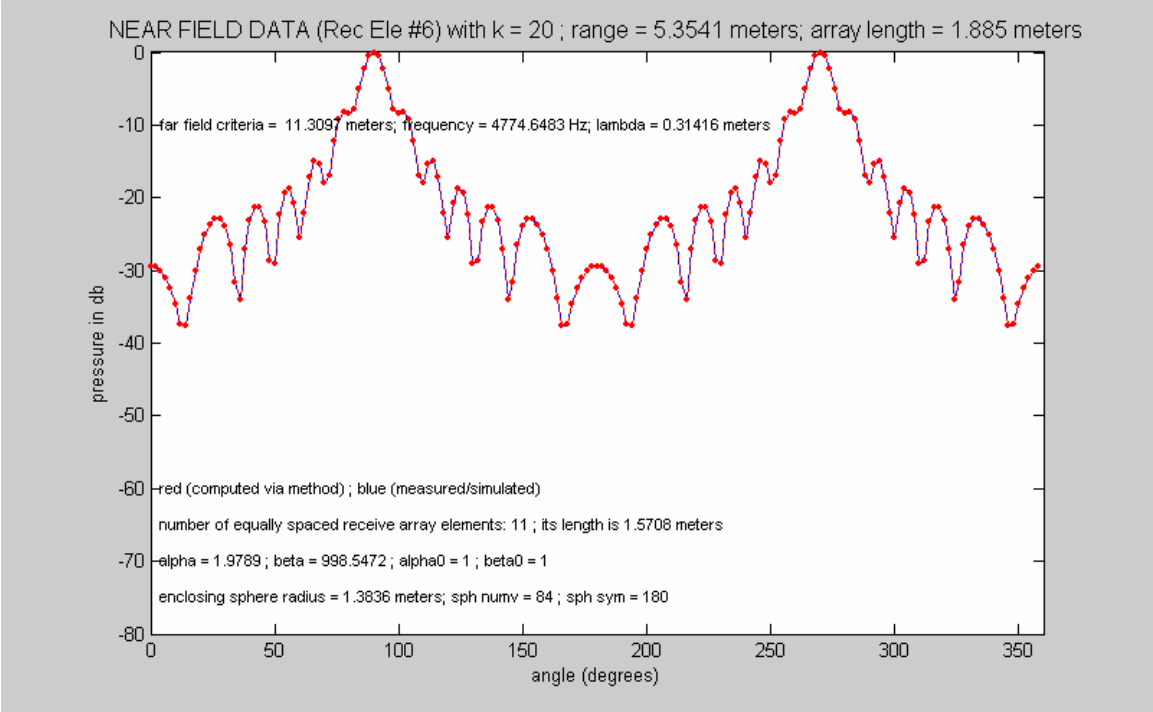


Figure 77 - Near Field Data – computed and simulated
alpha0=1; beta0=1; range error = -1 cm

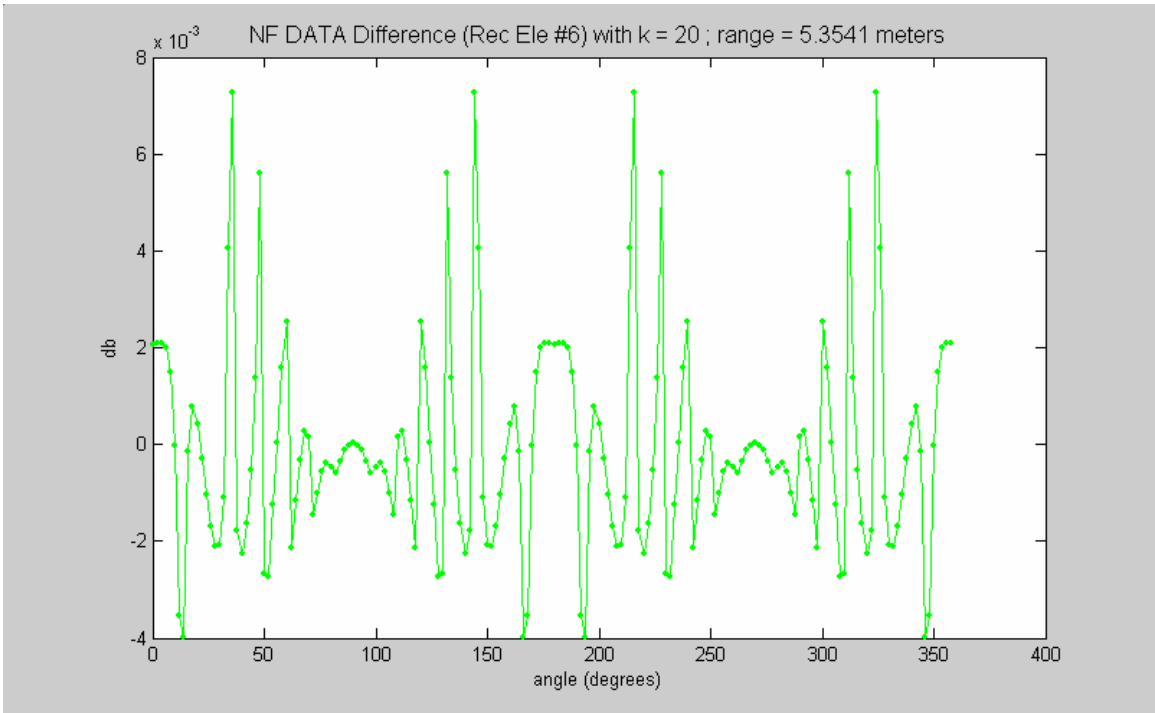


Figure 78 - Near Field Data Difference
alpha0=1; beta0=1; range error = -1 cm

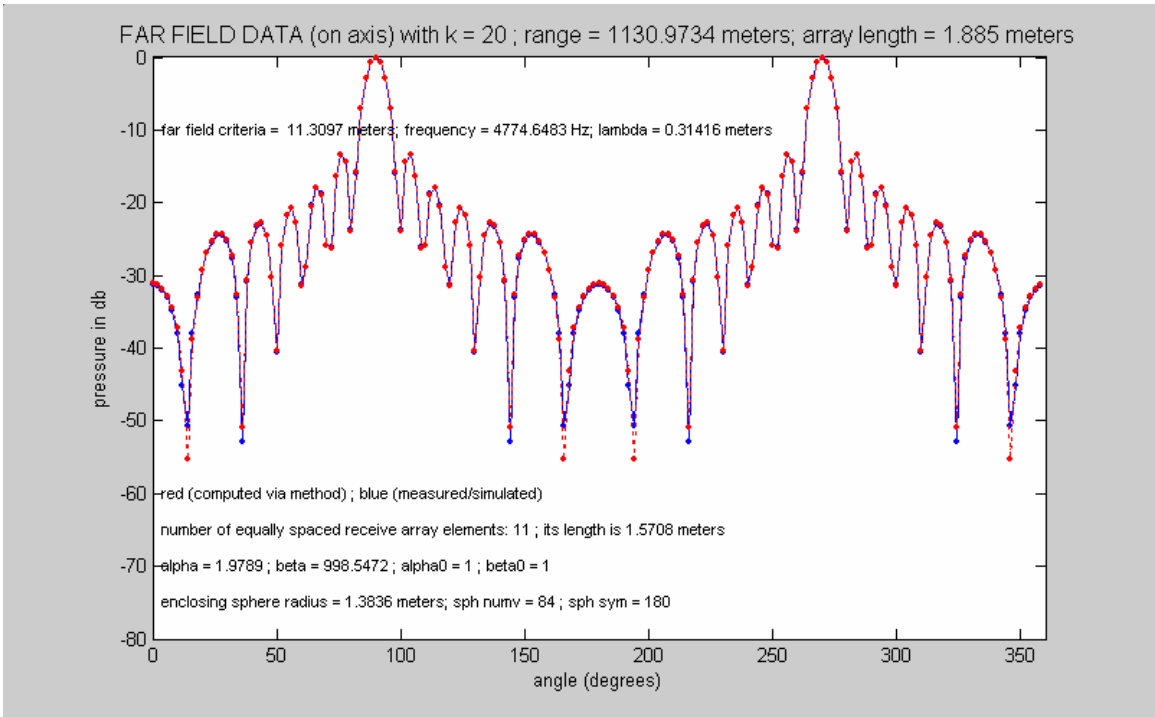


Figure 79 - Far Field Data – computed and simulated
alpha0=1; beta0=1; range error = -1 cm

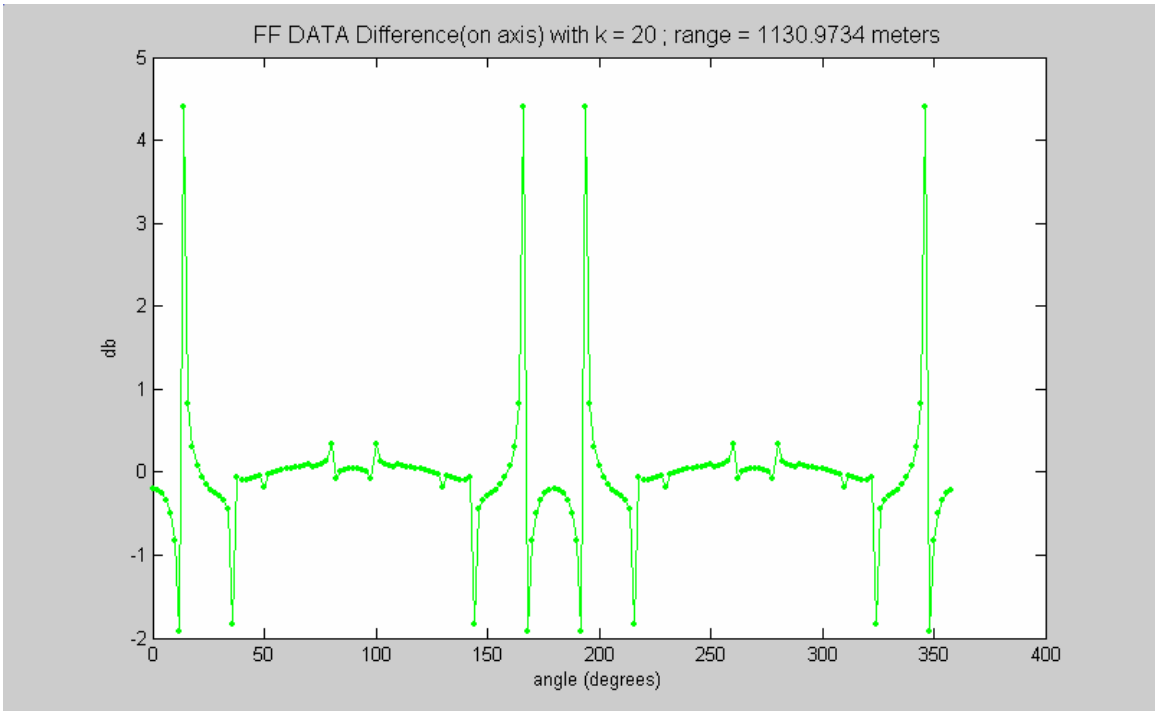


Figure 80 - Far Field Data Difference
alpha0=1; beta0=1; range error = -1 cm

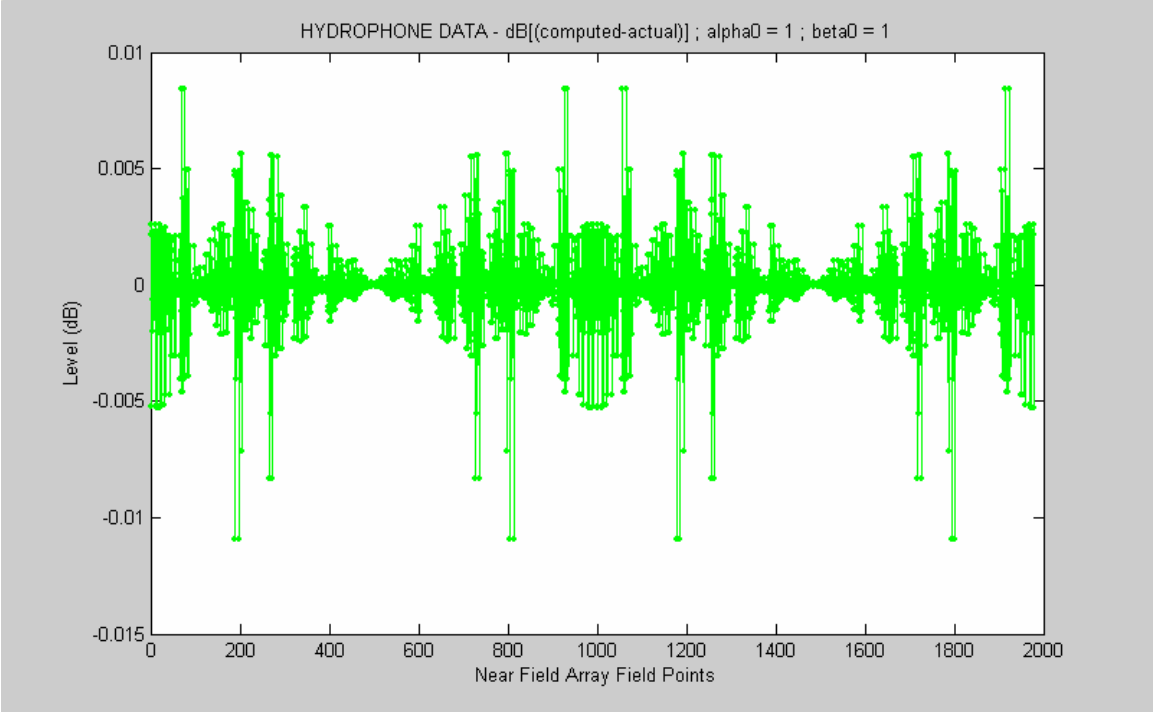


Figure 81 - Near Field Hydrophone Difference Data
alpha0=1; beta0=1; range error = +1 cm

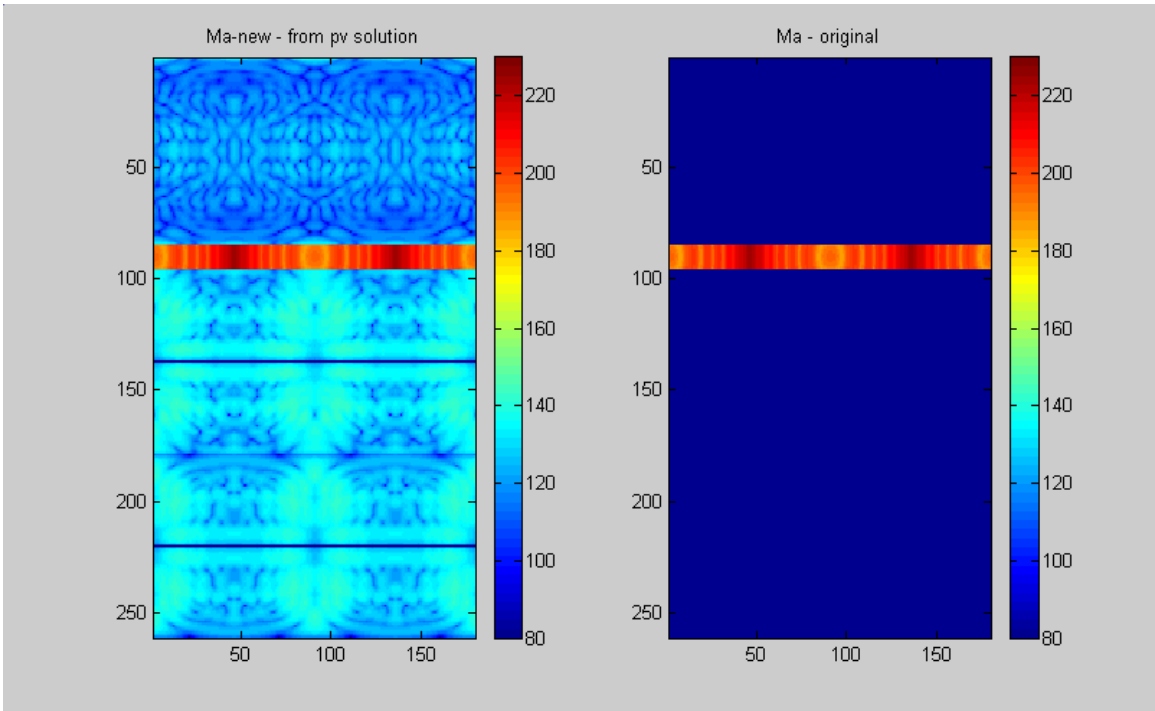


Figure 82 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0=1; range error = +1 cm

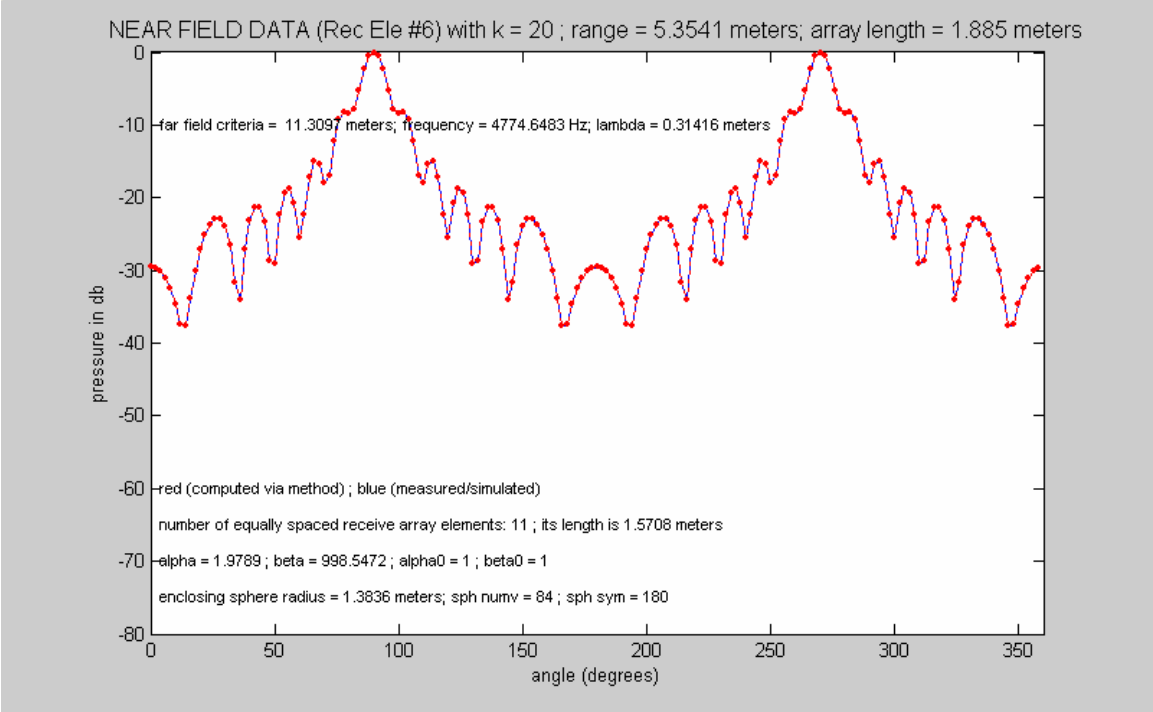


Figure 83 - Near Field Data – computed and simulated
alpha0=1; beta0=1; range error = +1 cm

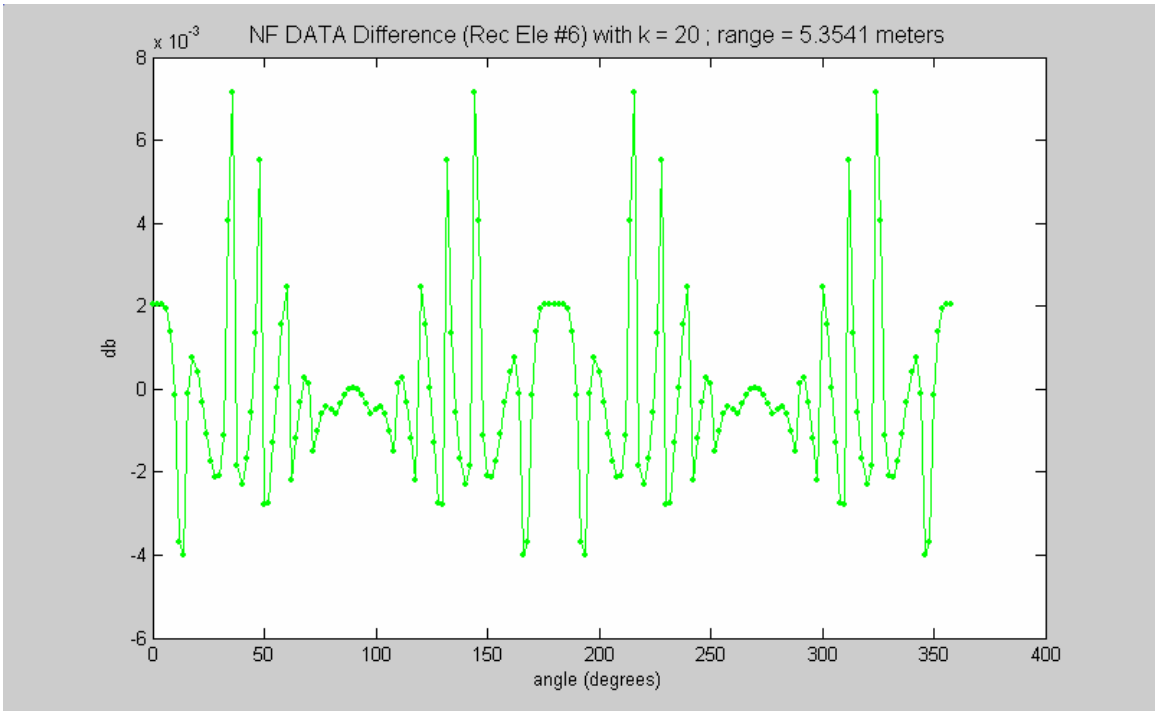


Figure 84 - Near Field Data Difference
alpha0=1; beta0=1; range error = +1 cm

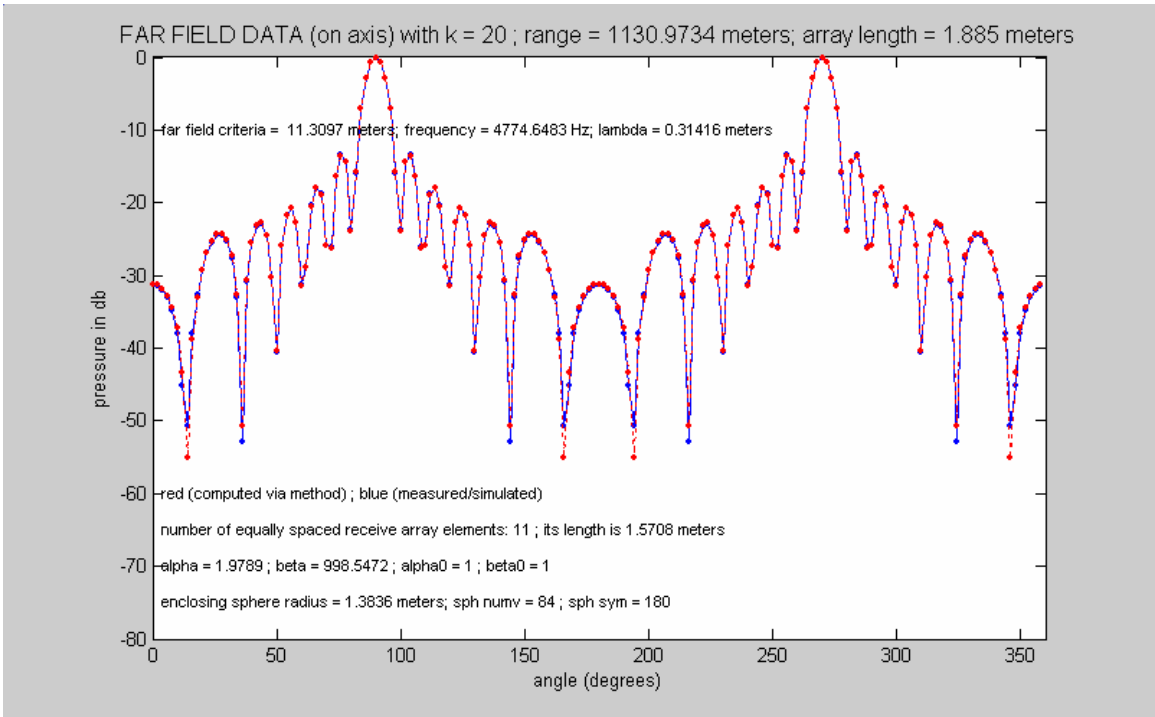


Figure 85 - Far Field Data – computed and simulated
alpha0=1; beta0=1; range error = +1 cm

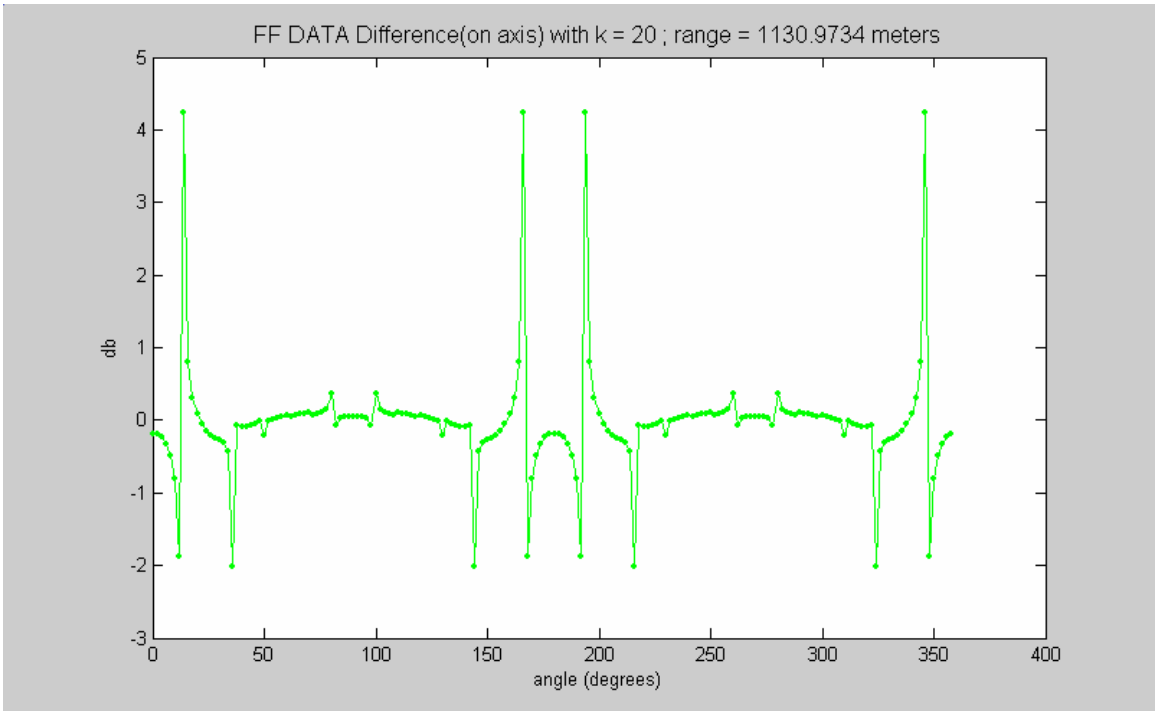


Figure 86 - Far Field Data Difference
alpha0=1; beta0=1; range error = +1 cm

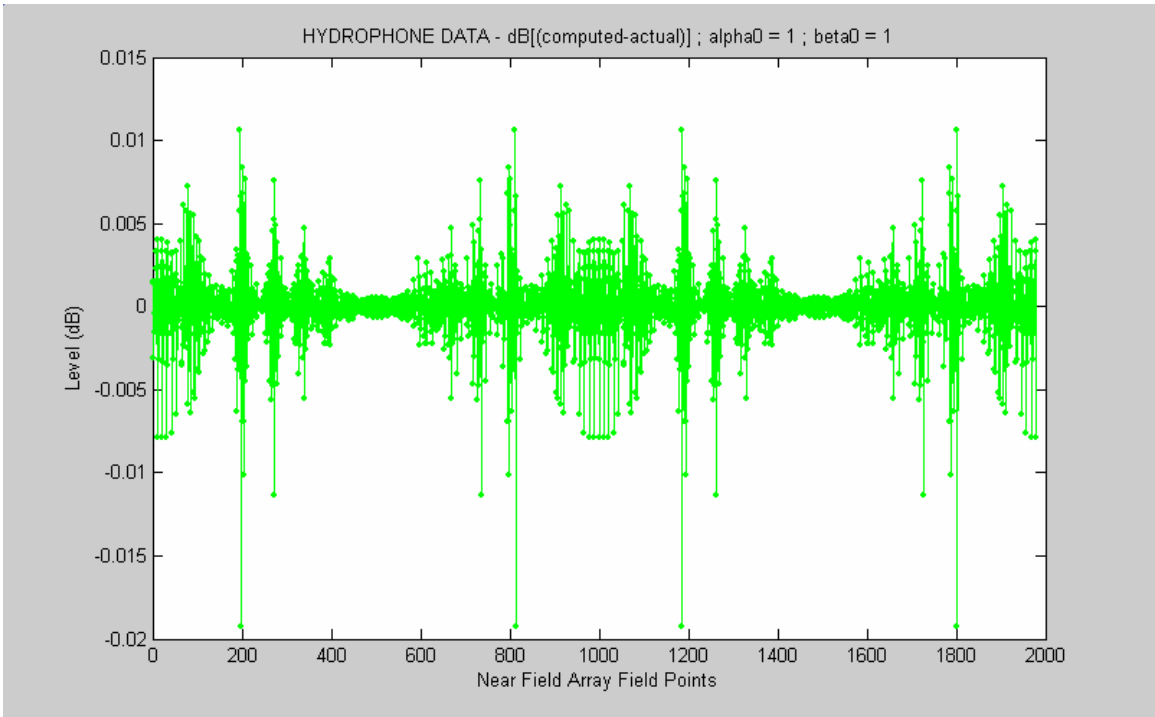


Figure 87 - Near Field Hydrophone Difference Data
alpha0=1; beta0=1; vertical receive array error = +lambda/2

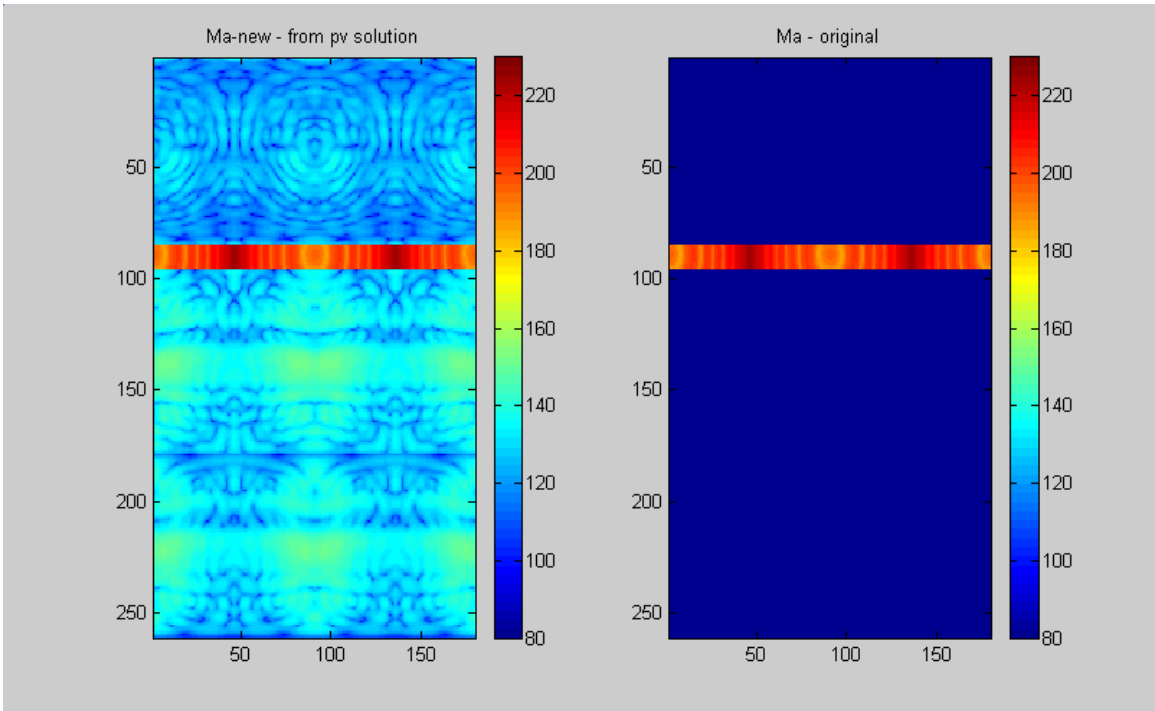


Figure 88 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0=1; vertical receive array error = +lambda/2

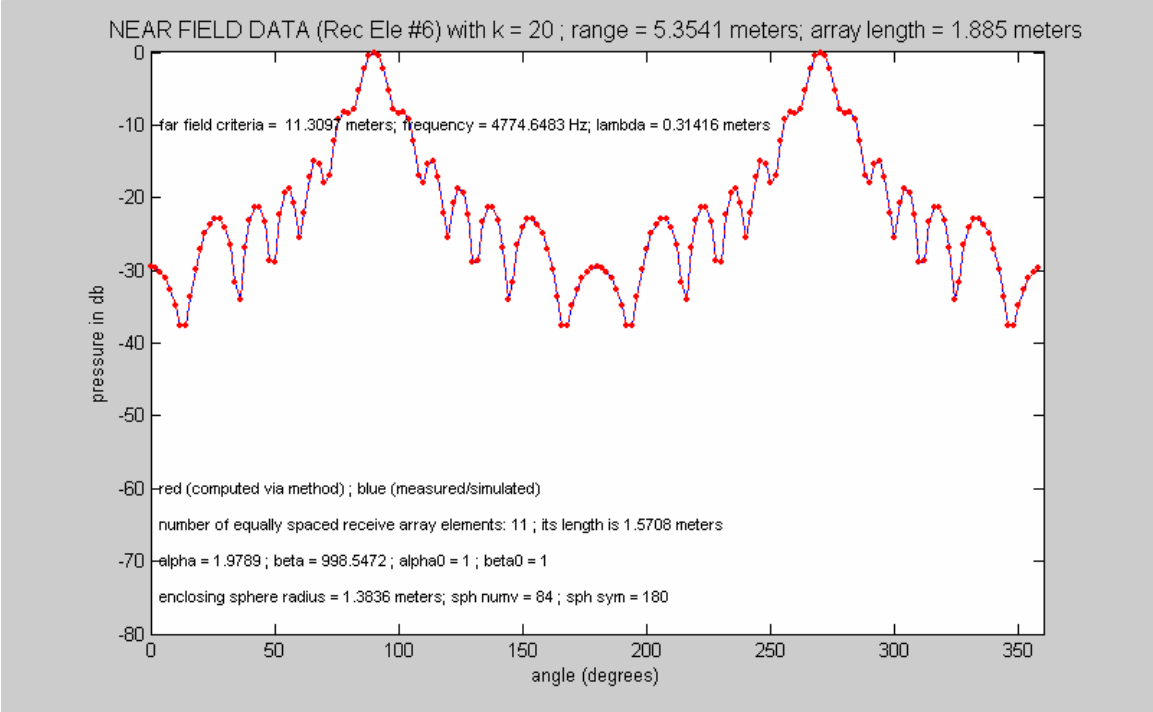


Figure 89 - Near Field Data – computed and simulated
alpha0=1; beta0=1; vertical receive array error = +lambda/2

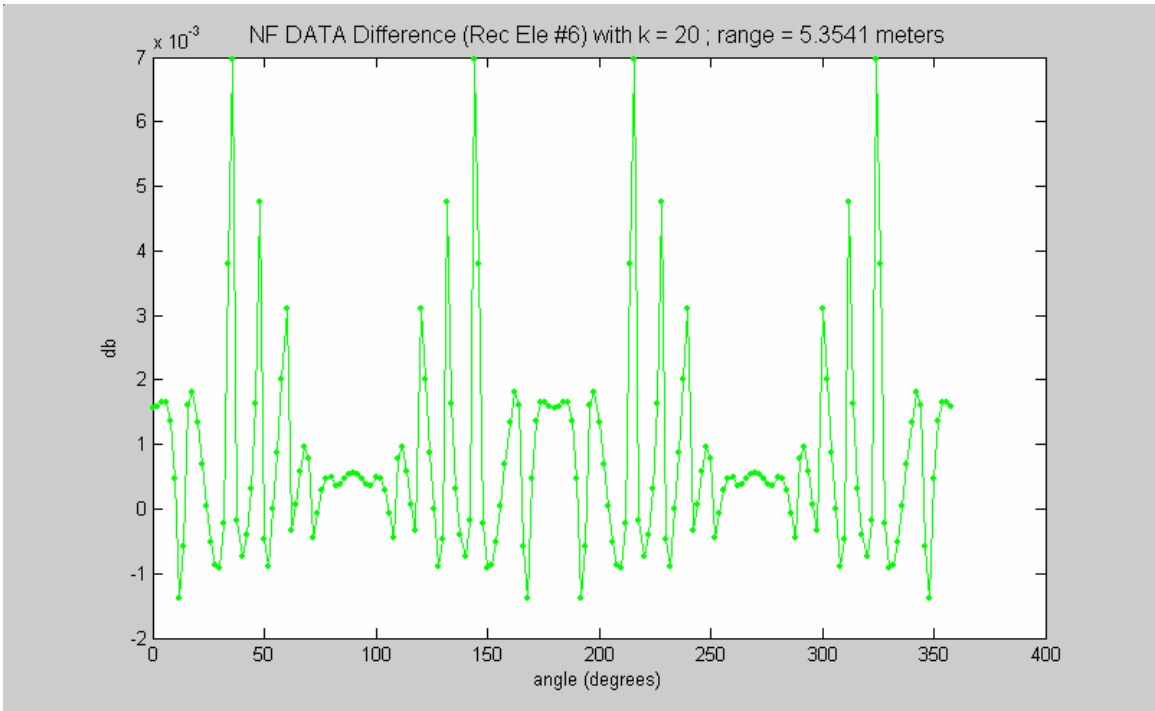


Figure 90 - Near Field Data Difference
alpha0=1; beta0=1; vertical receive array error = +lambda/2

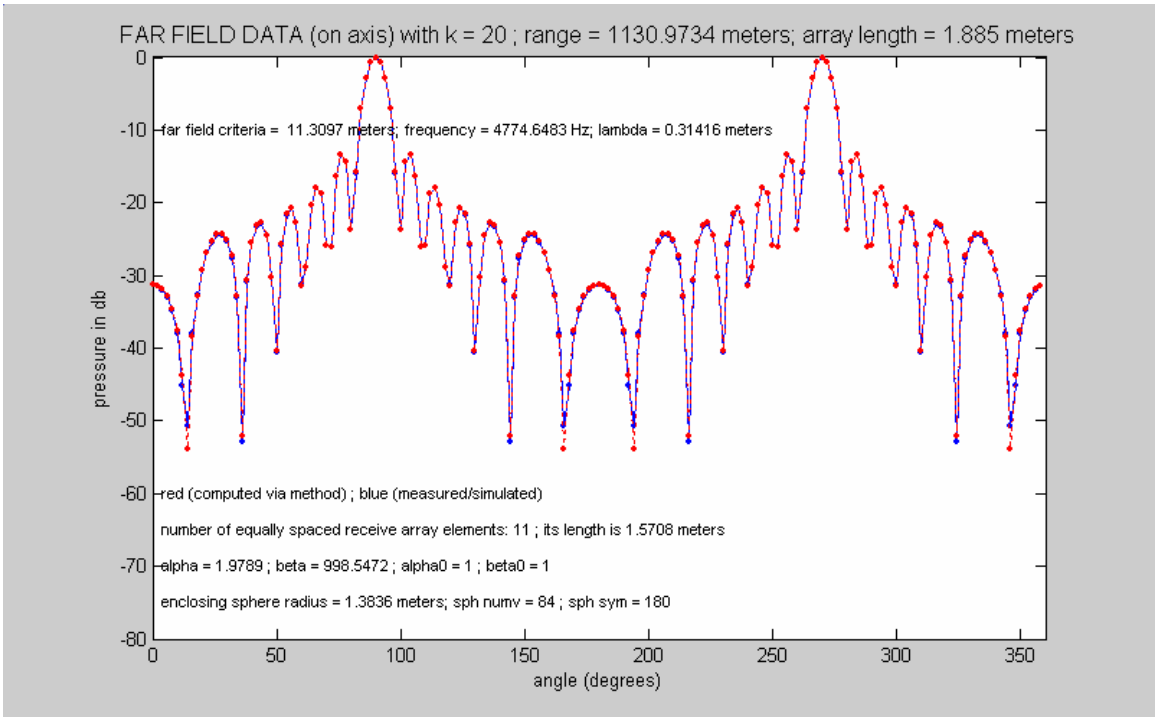


Figure 91 - Far Field Data – computed and simulated
alpha0=1; beta0=1; vertical receive array error = +lambda/2

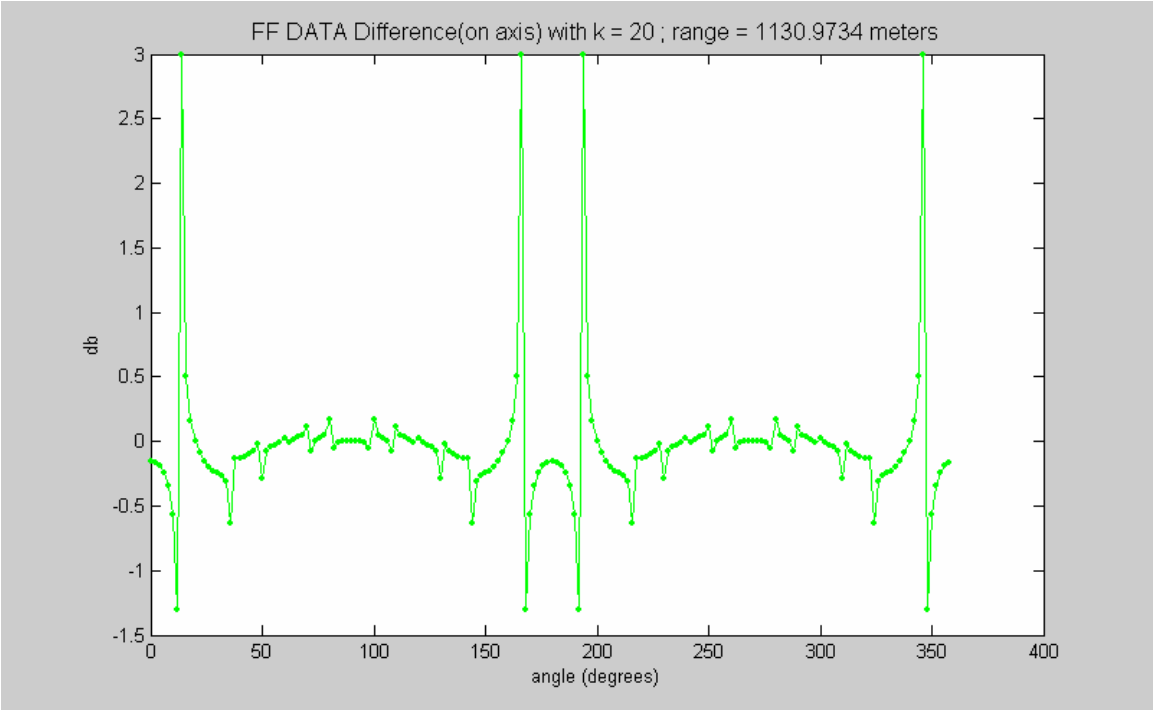


Figure 92 - Far Field Data Difference
alpha0=1; beta0=1; vertical receive array error = +lambda/2

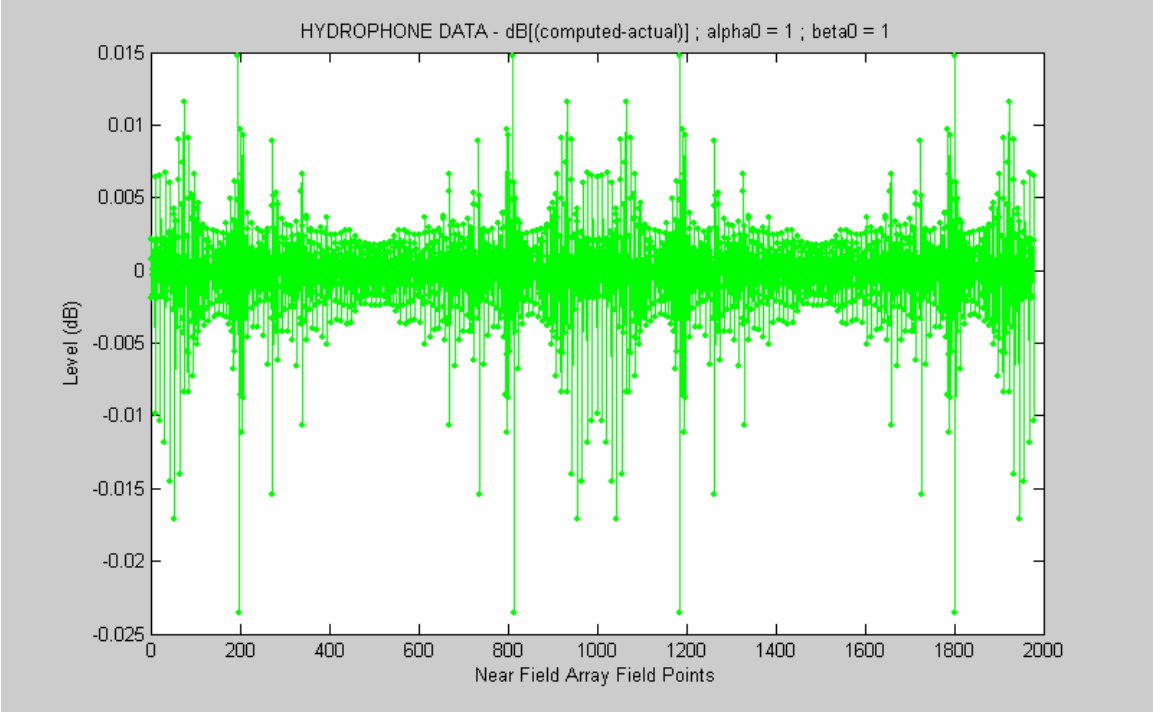


Figure 93 - Near Field Hydrophone Difference Data
alpha0=1; beta0=1; vertical receive array error = +lambda

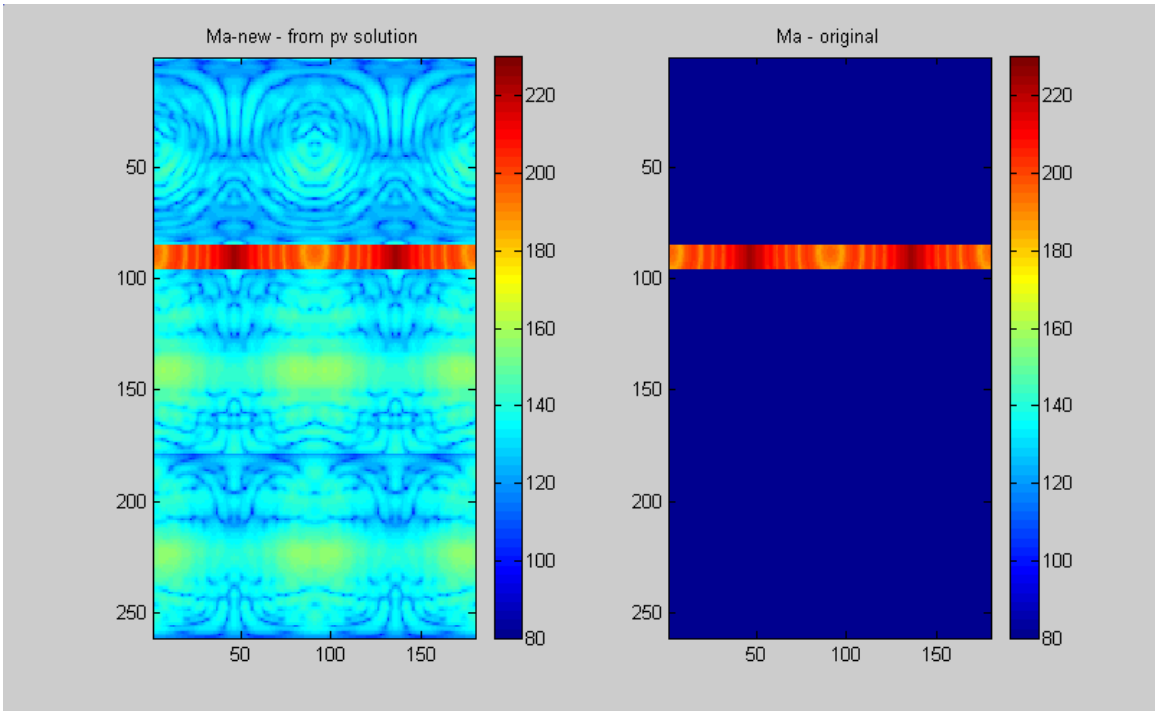


Figure 94 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0=1; vertical receive array error = +lambda

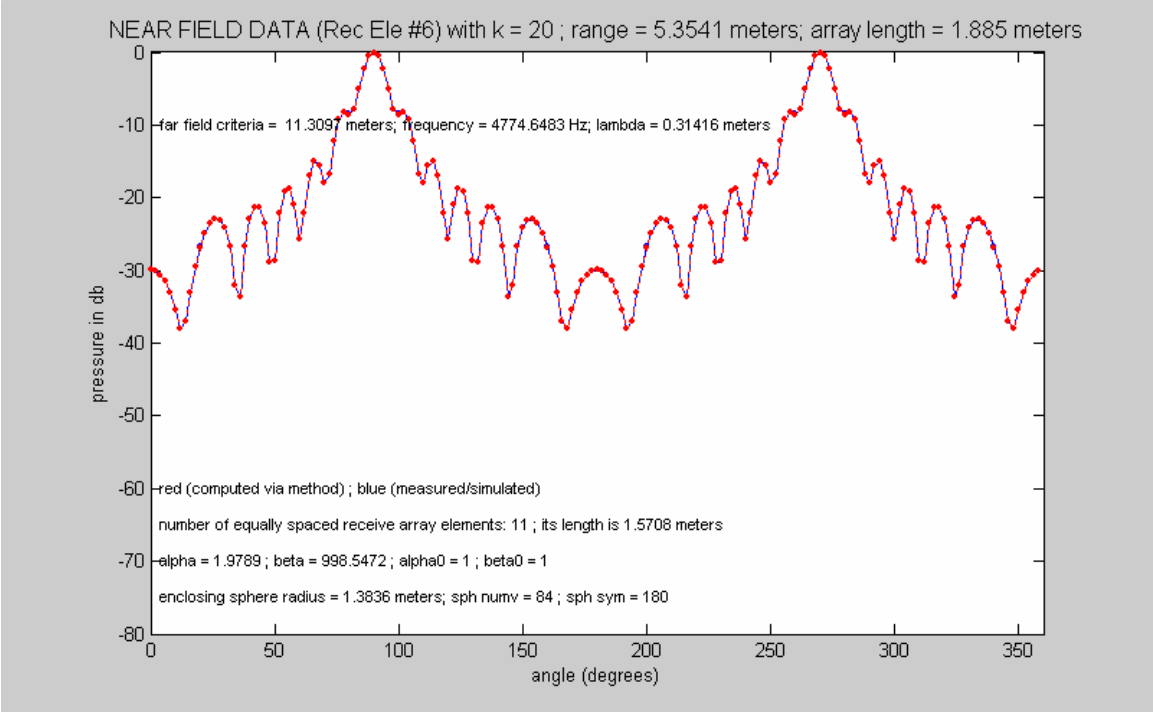


Figure 95 - Near Field Data – computed and simulated
alpha0=1; beta0=1; vertical receive array error = +lambda

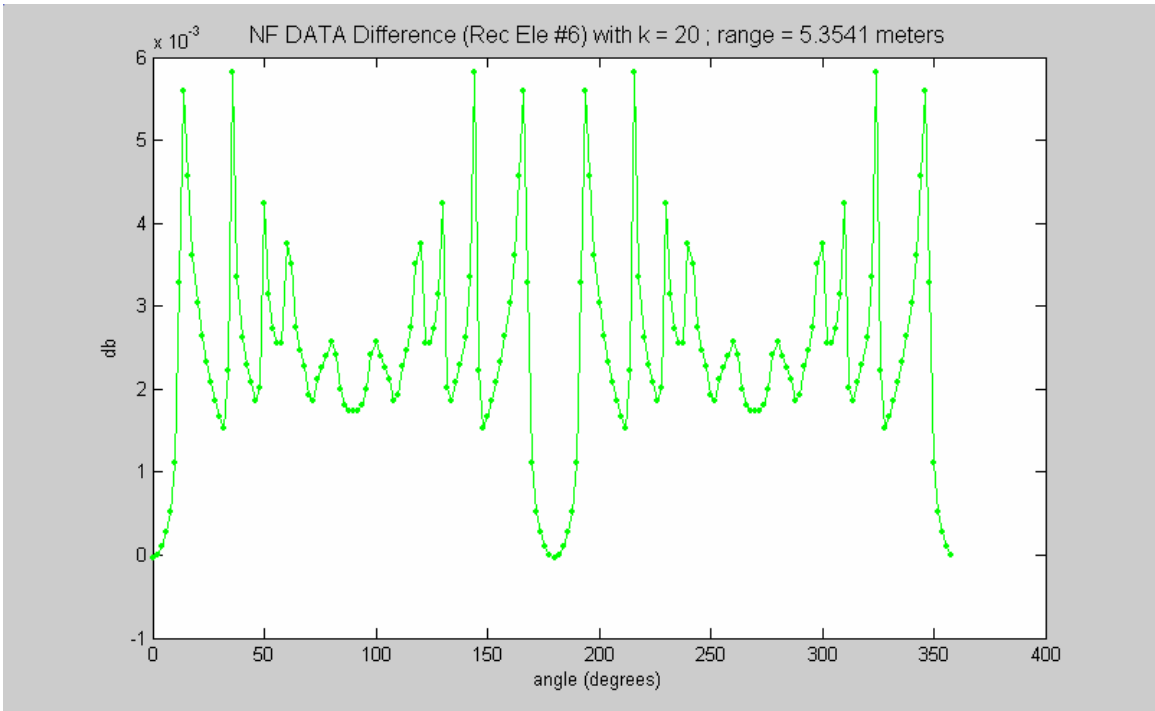


Figure 96 - Near Field Data Difference
alpha0=1; beta0=1; vertical receive array error = +lambda

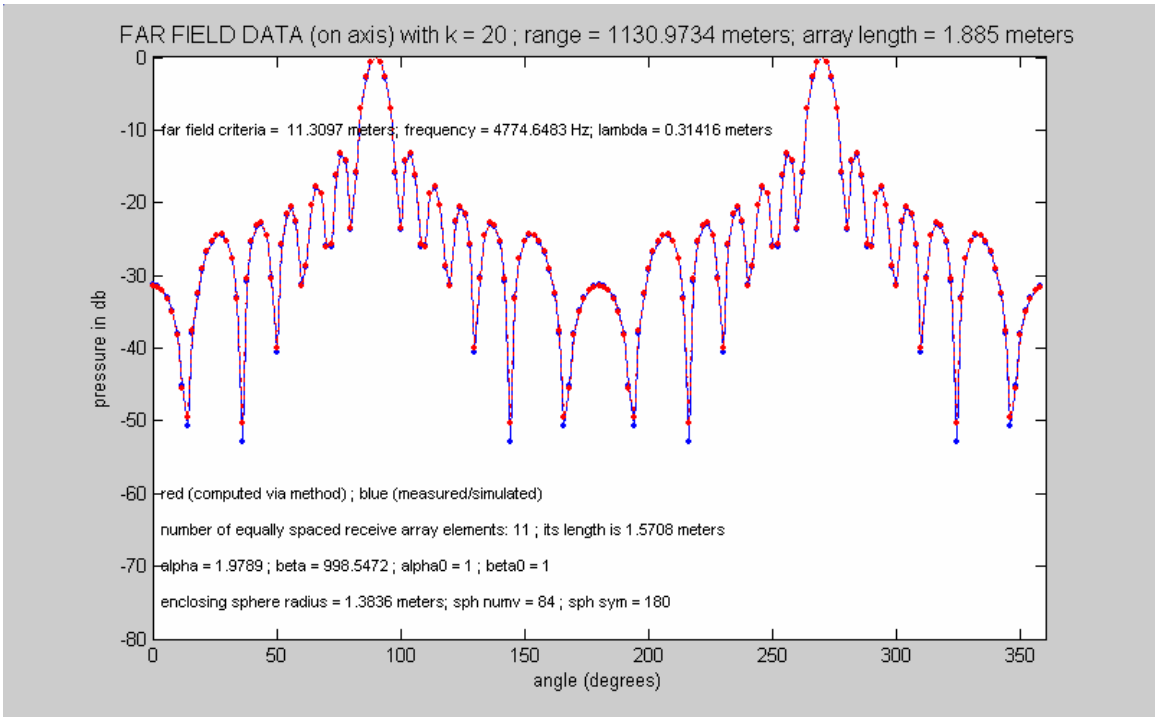


Figure 97 - Far Field Data – computed and simulated
alpha0=1; beta0=1; vertical receive array error = +lambda

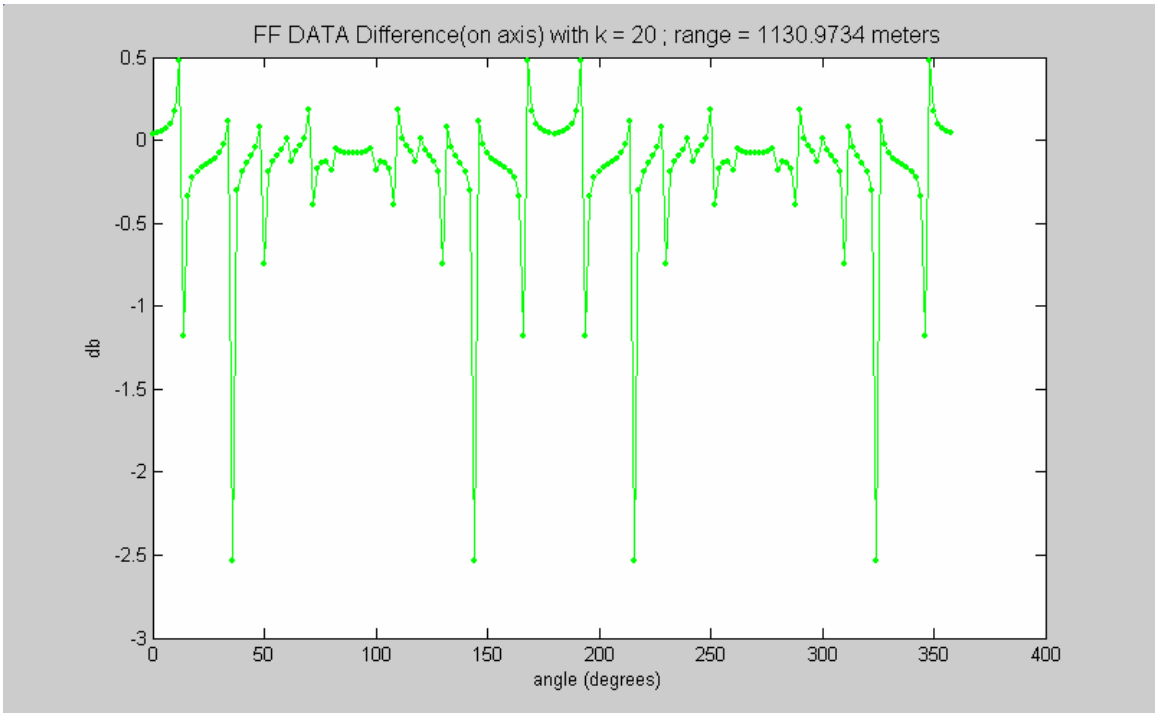


Figure 98 - Far Field Data Difference
alpha0=1; beta0=1; vertical receive array error = +lambda

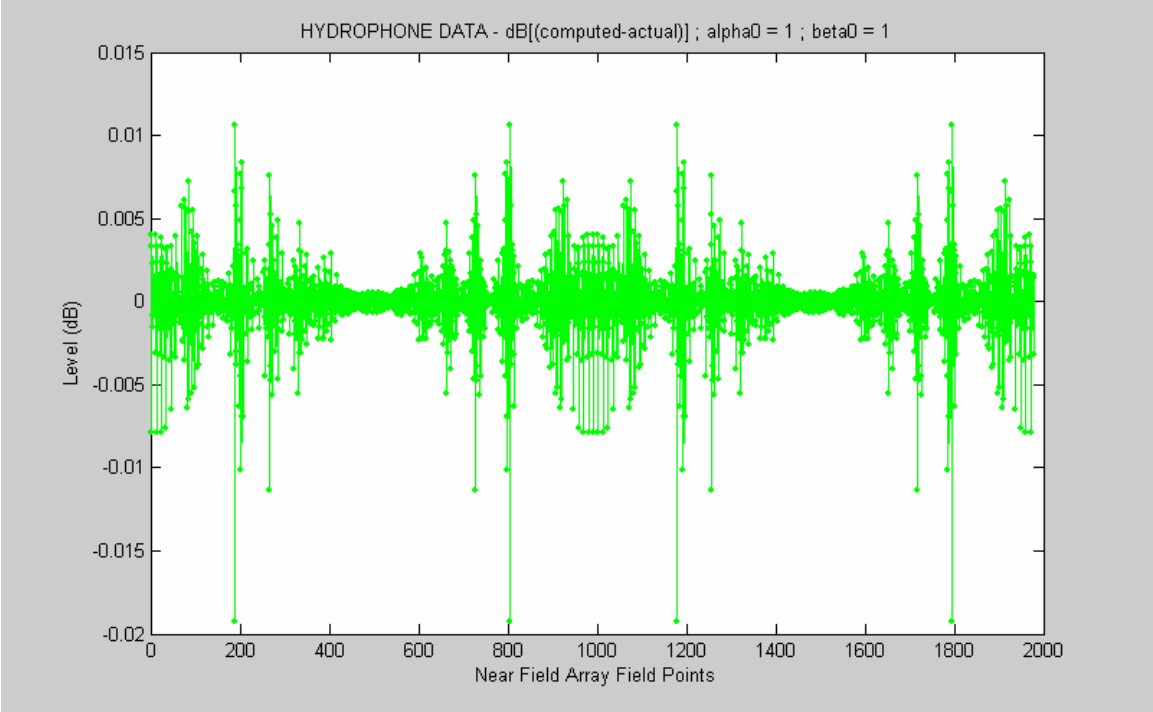


Figure 99 - Near Field Hydrophone Difference Data
alpha0=1; beta0=1; vertical receive array error = -lambda/2

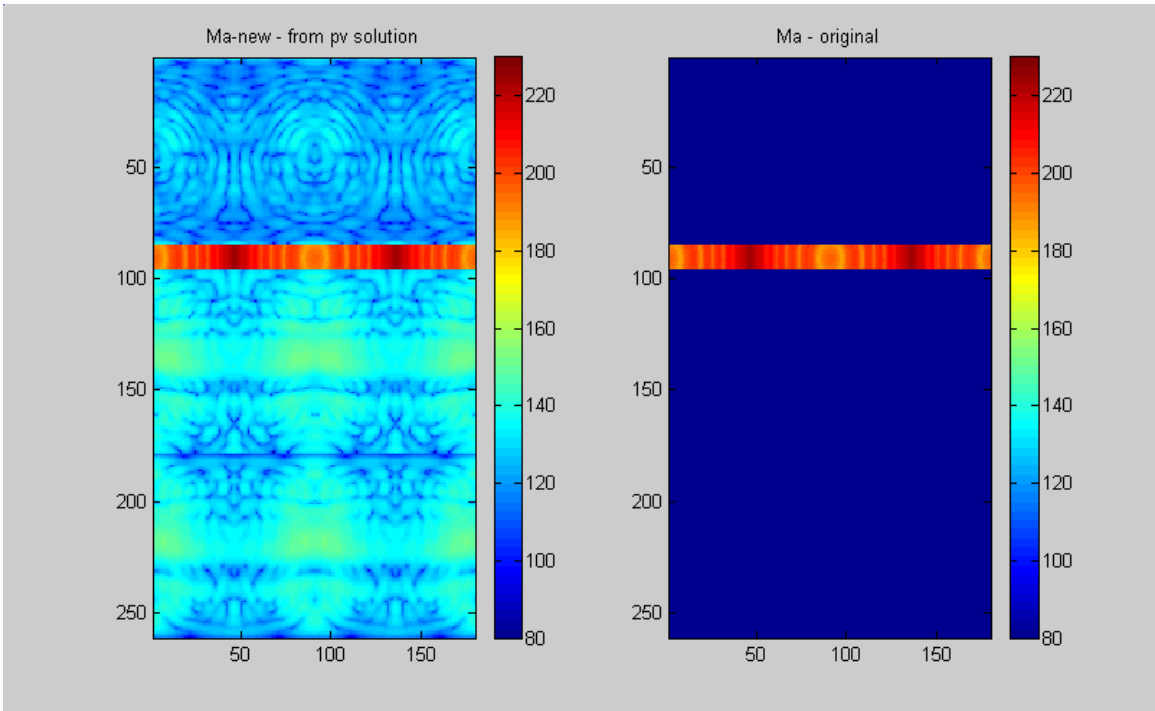


Figure 100 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0=1; vertical receive array error = -lambda/2

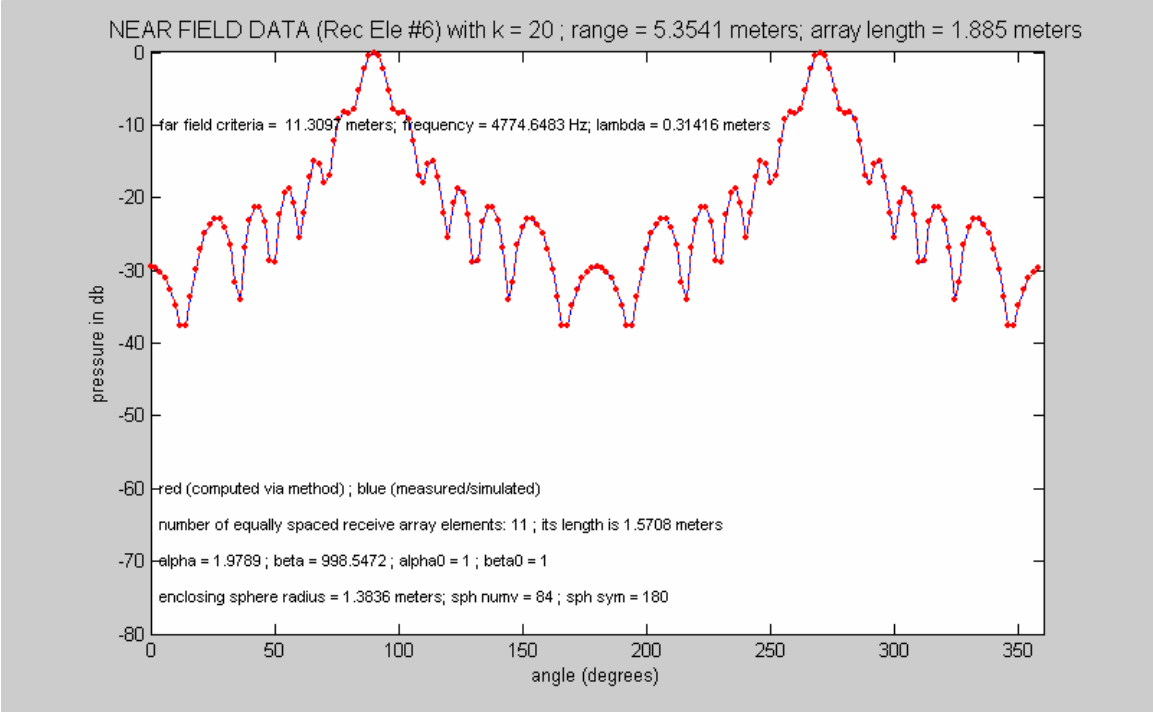


Figure 101 - Near Field Data – computed and simulated
alpha0=1; beta0=1; vertical receive array error = -lambda/2

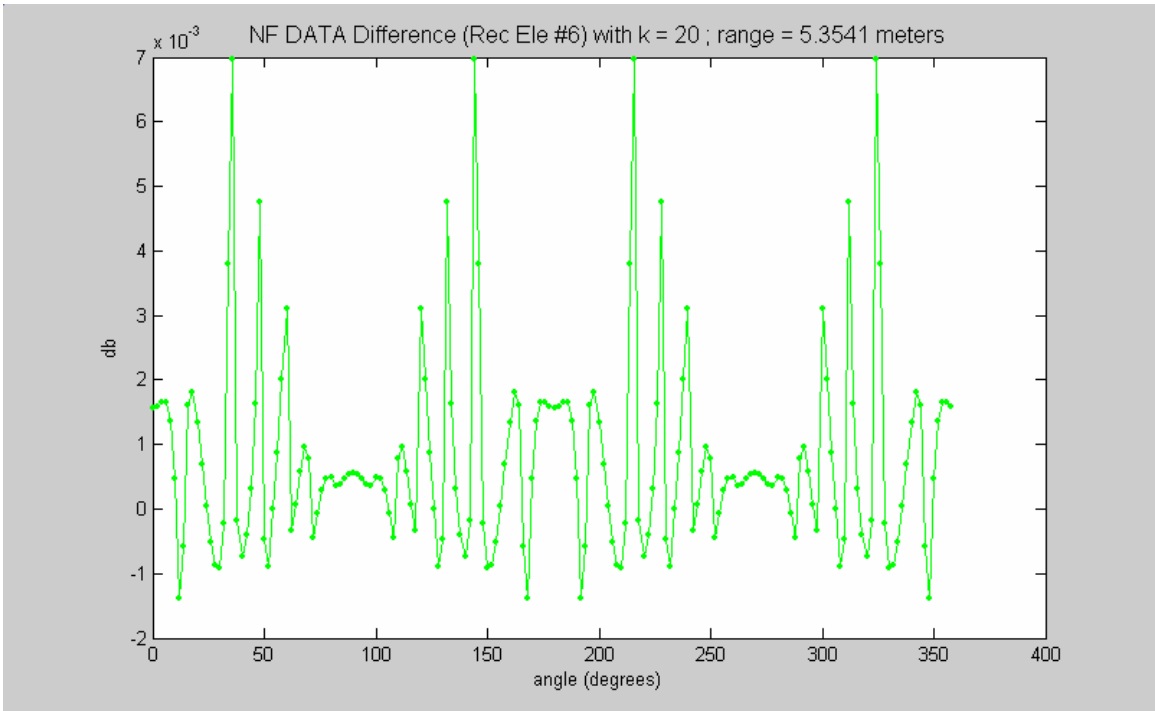


Figure 102 - Near Field Data Difference
alpha0=1; beta0=1; vertical receive array error = -lambda/2

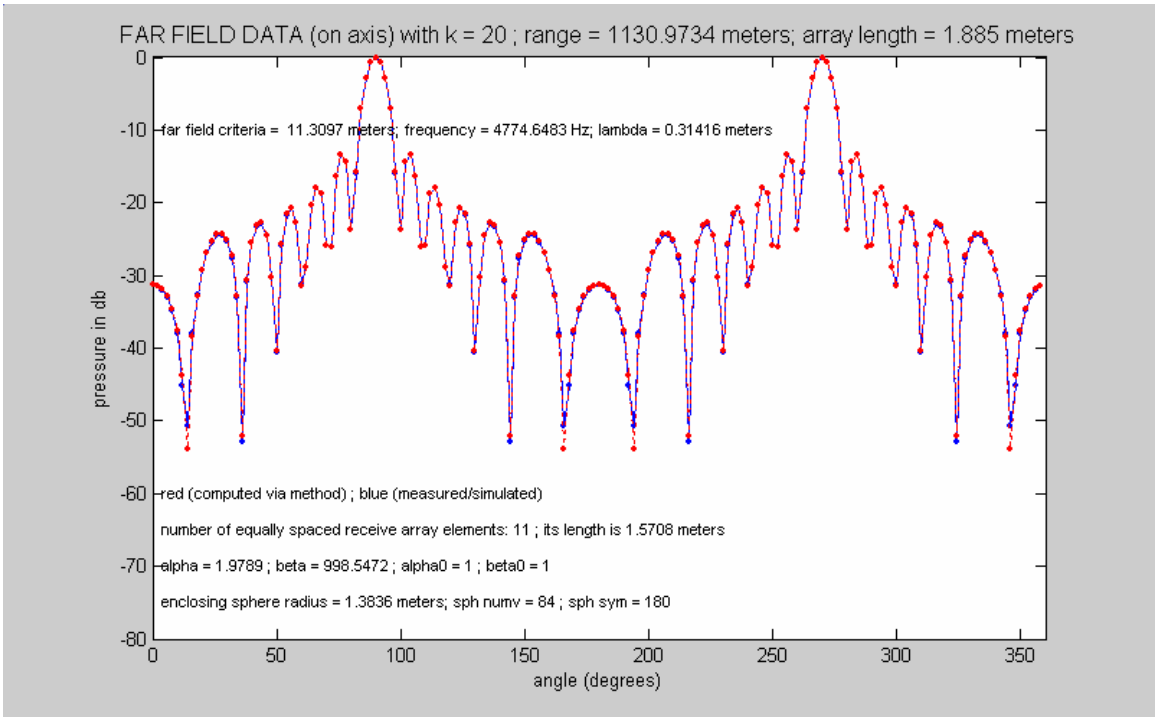


Figure 103 - Far Field Data – computed and simulated
alpha0=1; beta0=1; vertical receive array error = -lambda/2

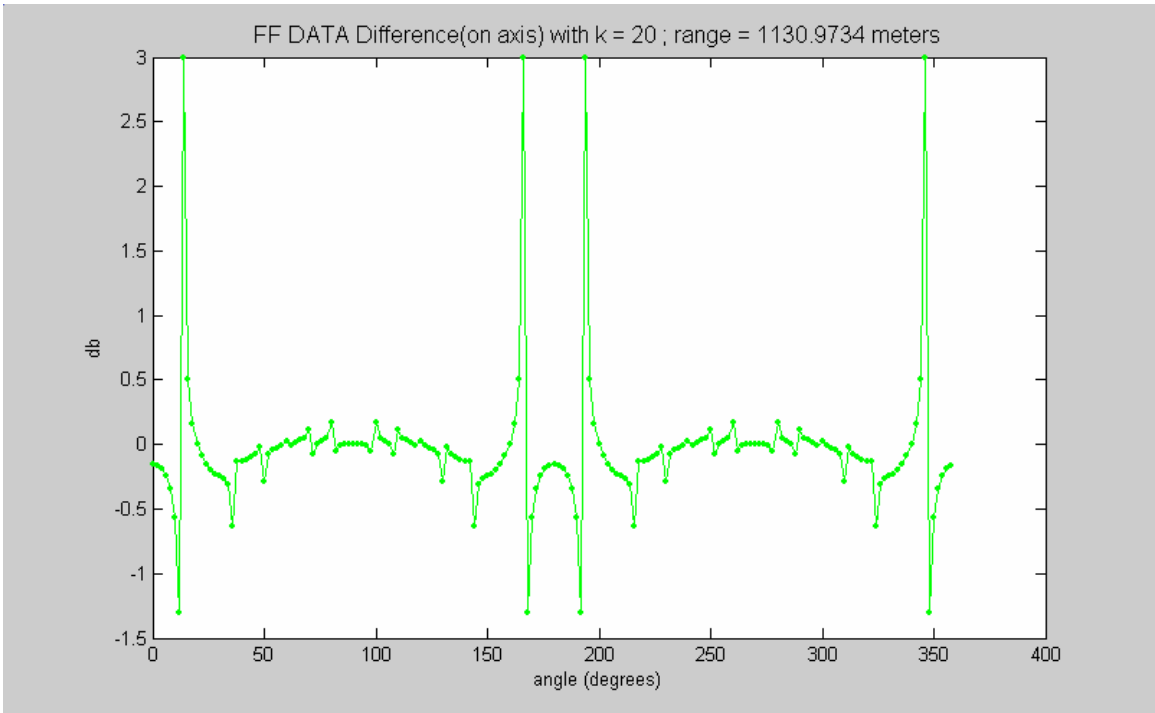


Figure 104 - Far Field Data Difference
alpha0=1; beta0=1; vertical receive array error = -lambda/2

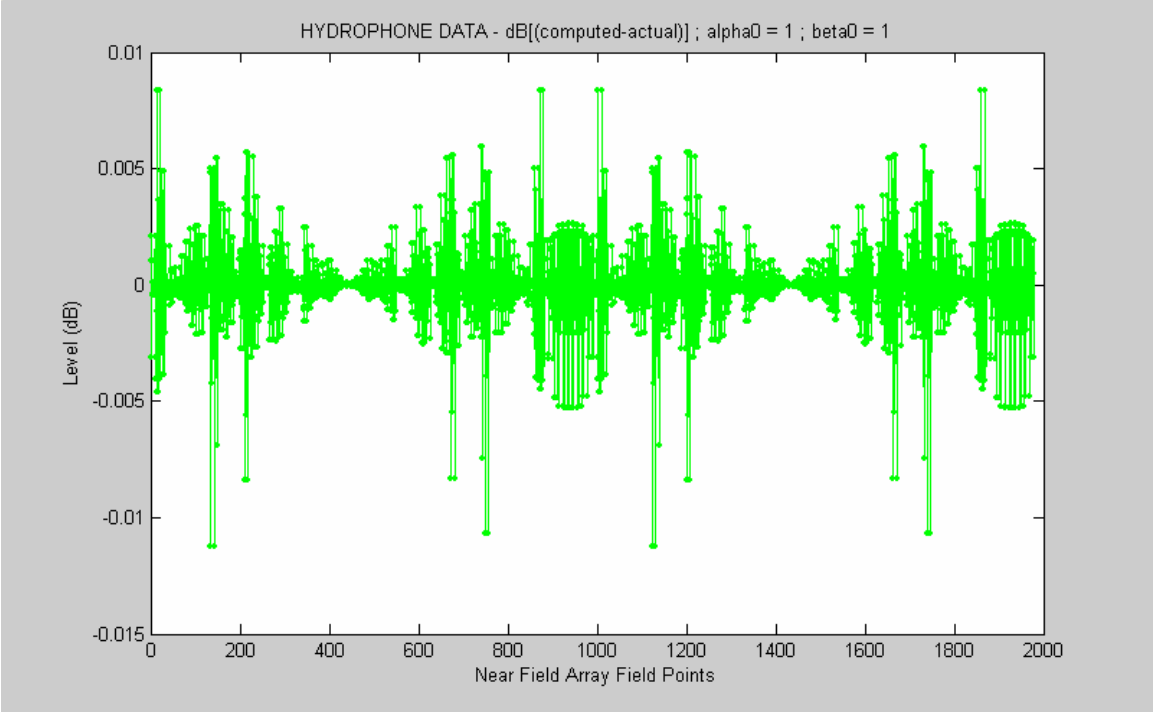


Figure 105 - Near Field Hydrophone Difference Data
alpha0=1; beta0=1; azimuthal error = +10 degrees

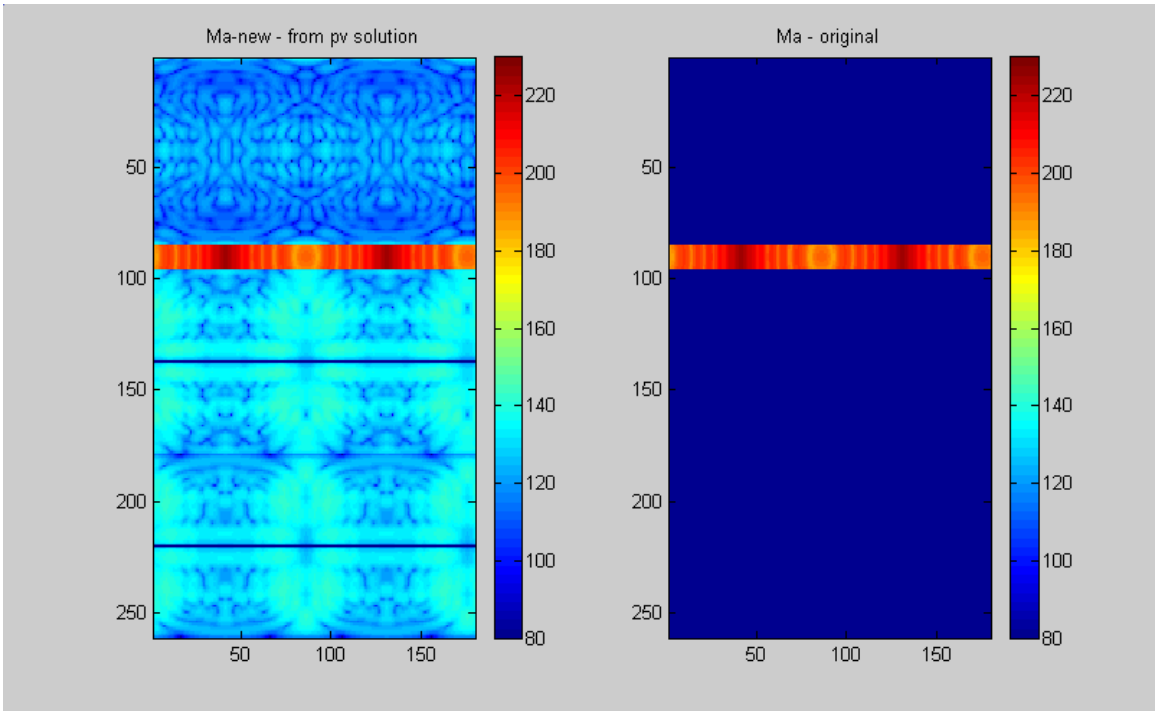


Figure 106 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
alpha0=1; beta0=1; azimuthal error = +10 degrees

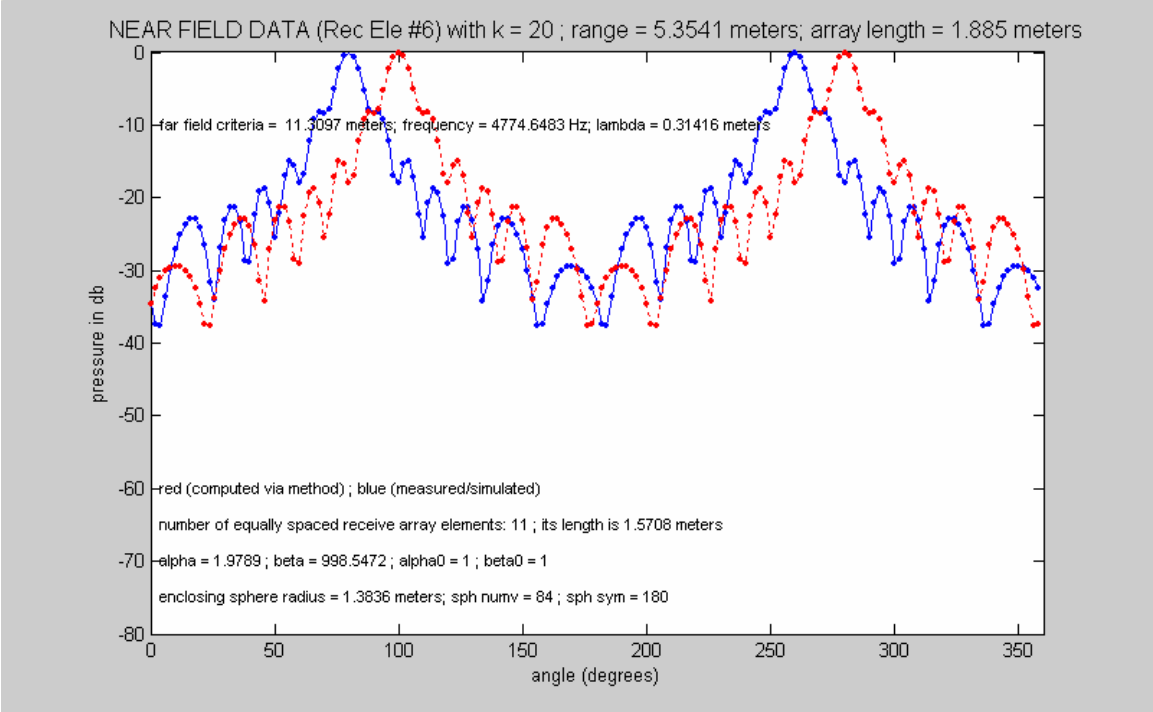


Figure 107 - Near Field Data – computed and simulated
alpha0=1; beta0=1; azimuthal error = +10 degrees

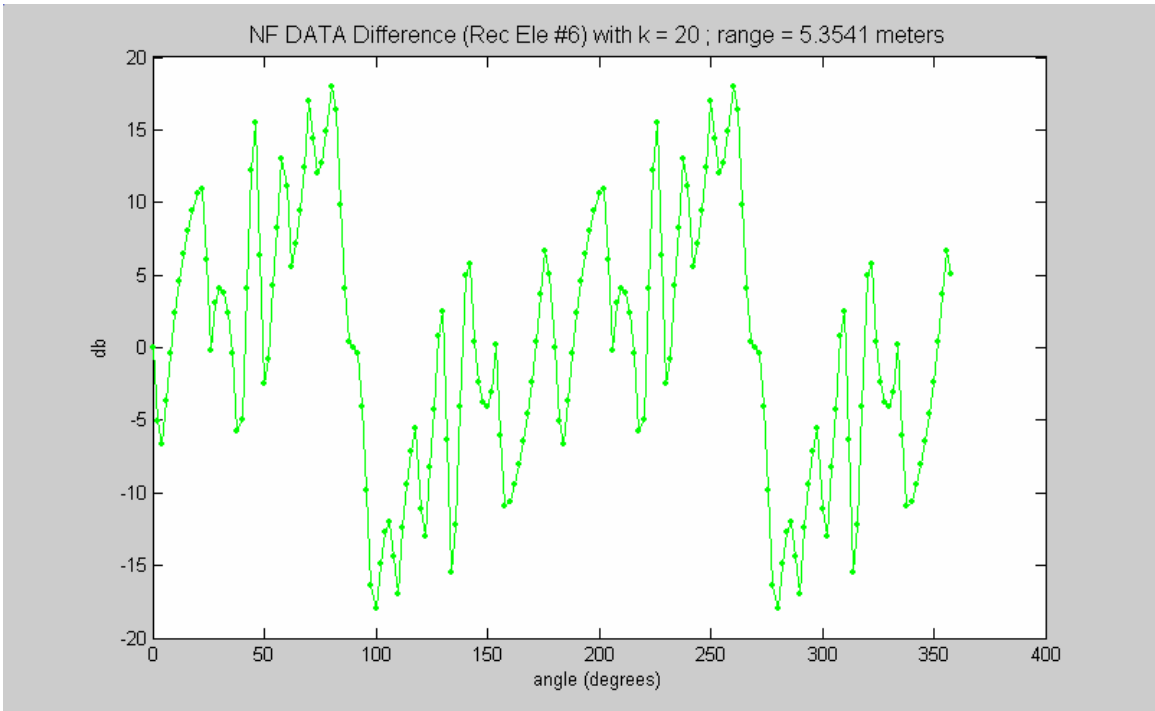


Figure 108 - Near Field Data Difference
alpha0=1; beta0=1; azimuthal error = +10 degrees

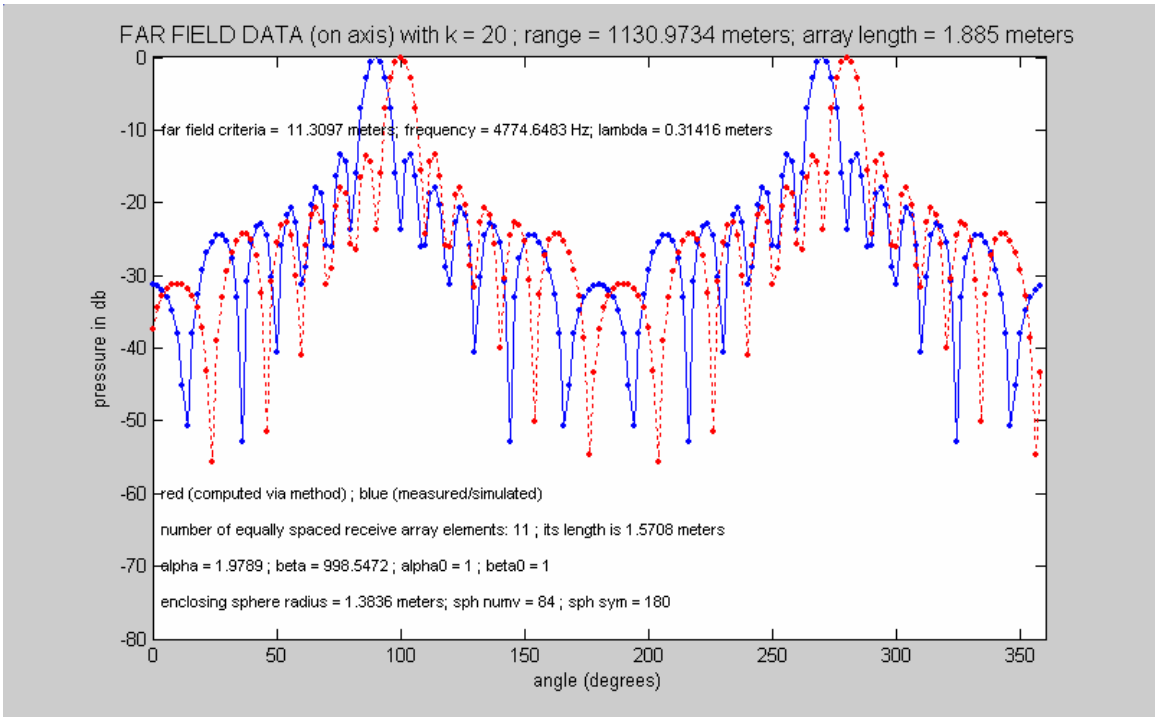


Figure 109 - Far Field Data – computed and simulated
alpha0=1; beta0=1; azimuthal error = +10 degrees

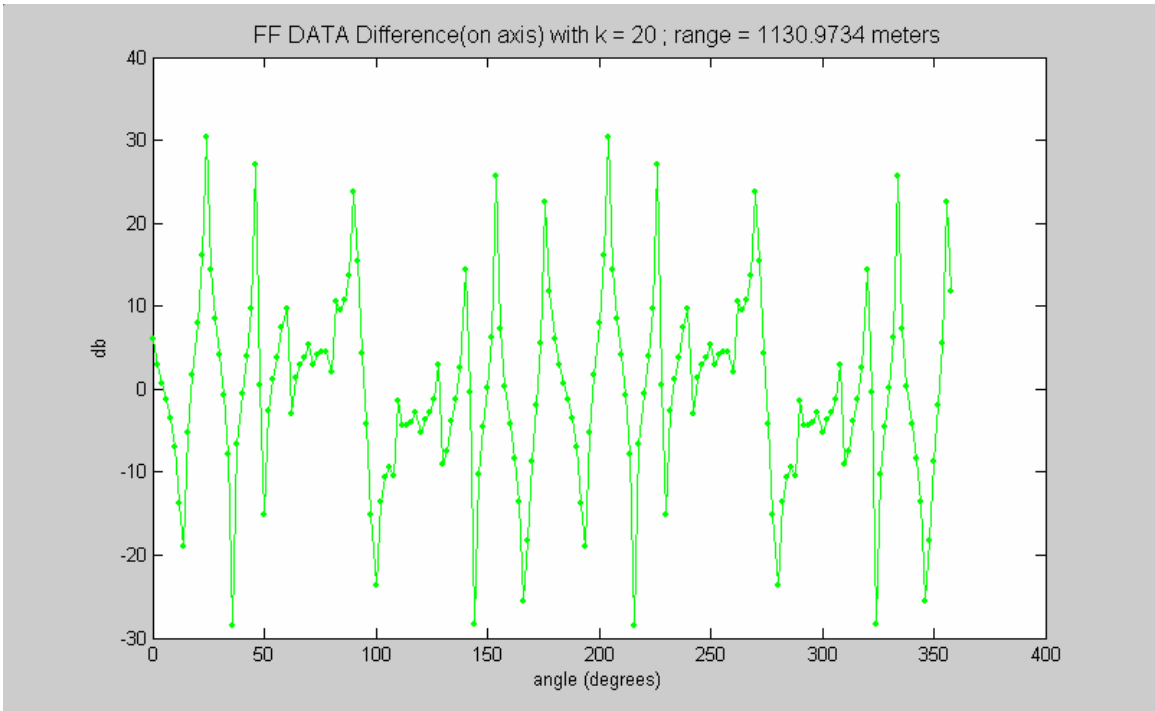


Figure 110 - Far Field Data Difference
alpha0=1; beta0=1; azimuthal error = +10 degrees

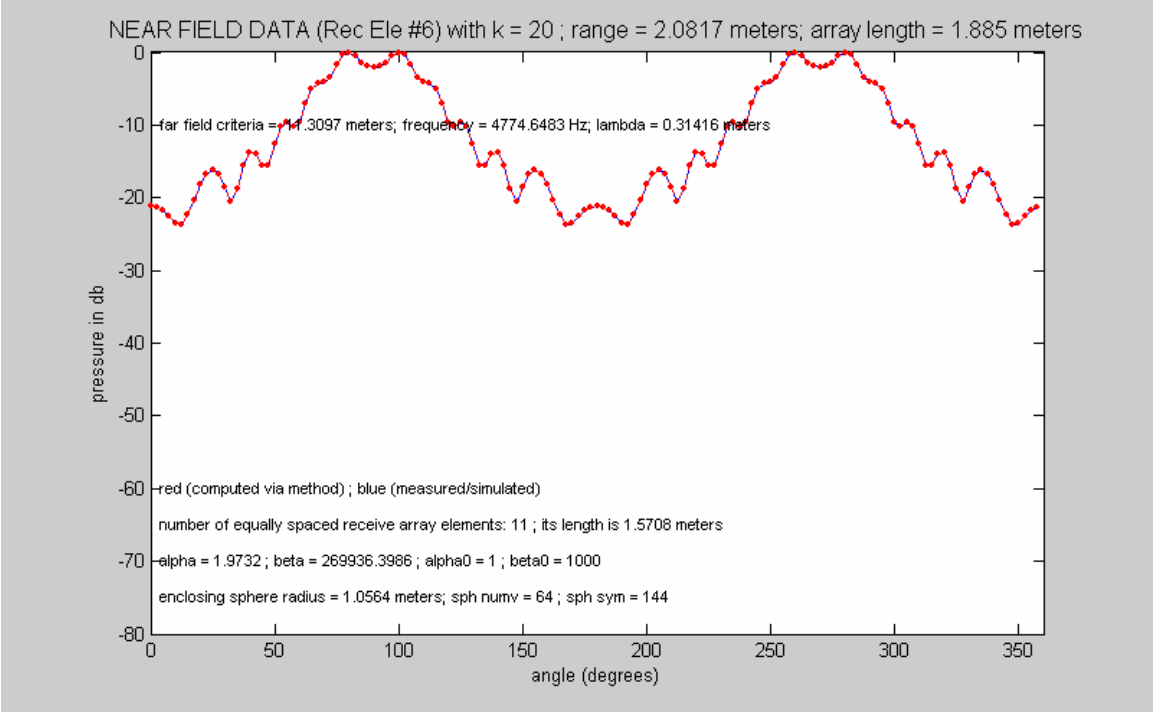


Figure 111 - Near Field Data – computed and simulated
Using 2.082 meter range, 1.571 meter long, 11 element near field array data

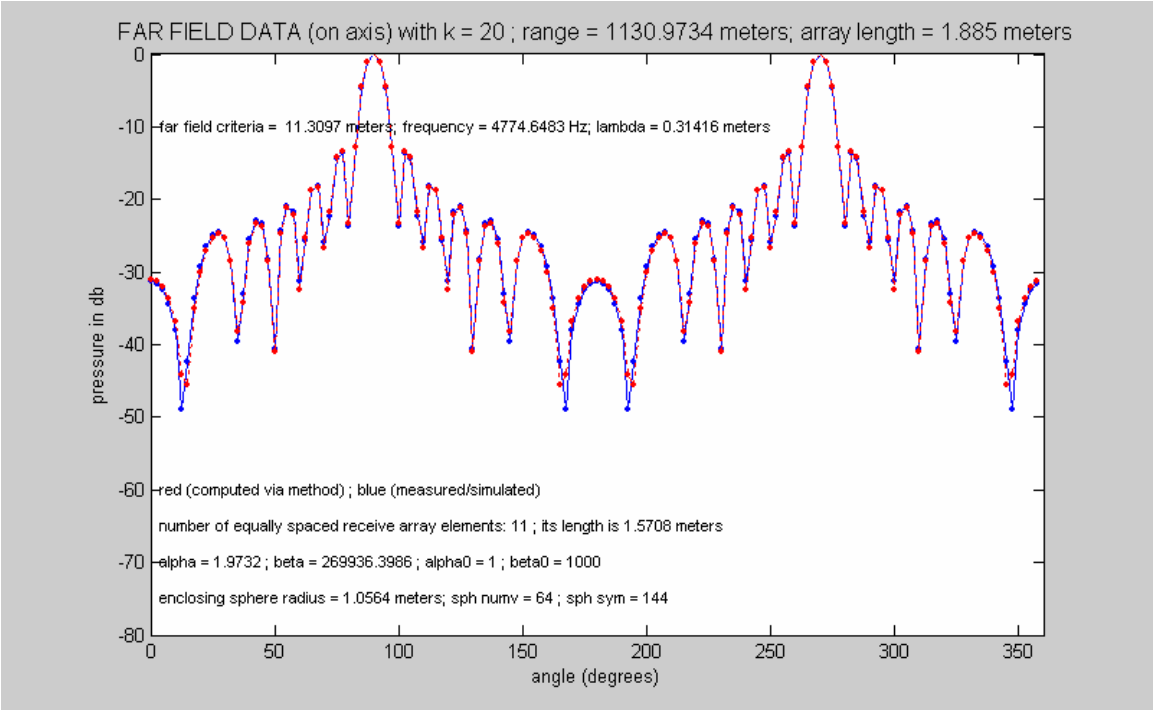


Figure 112 - Far Field Data – computed and simulated
Using 2.082 meter range, 1.571 meter long, 11 element near field array data

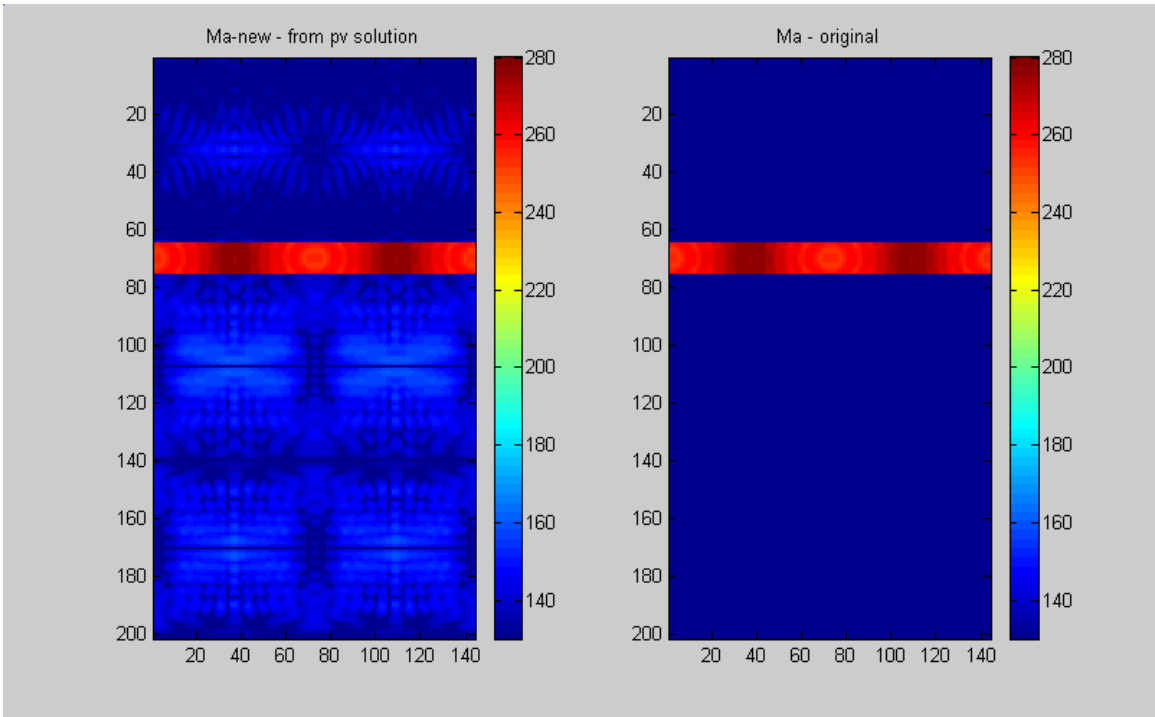


Figure 113 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
Using 2.082 meter range, 1.571 meter long, 11 element near field array data

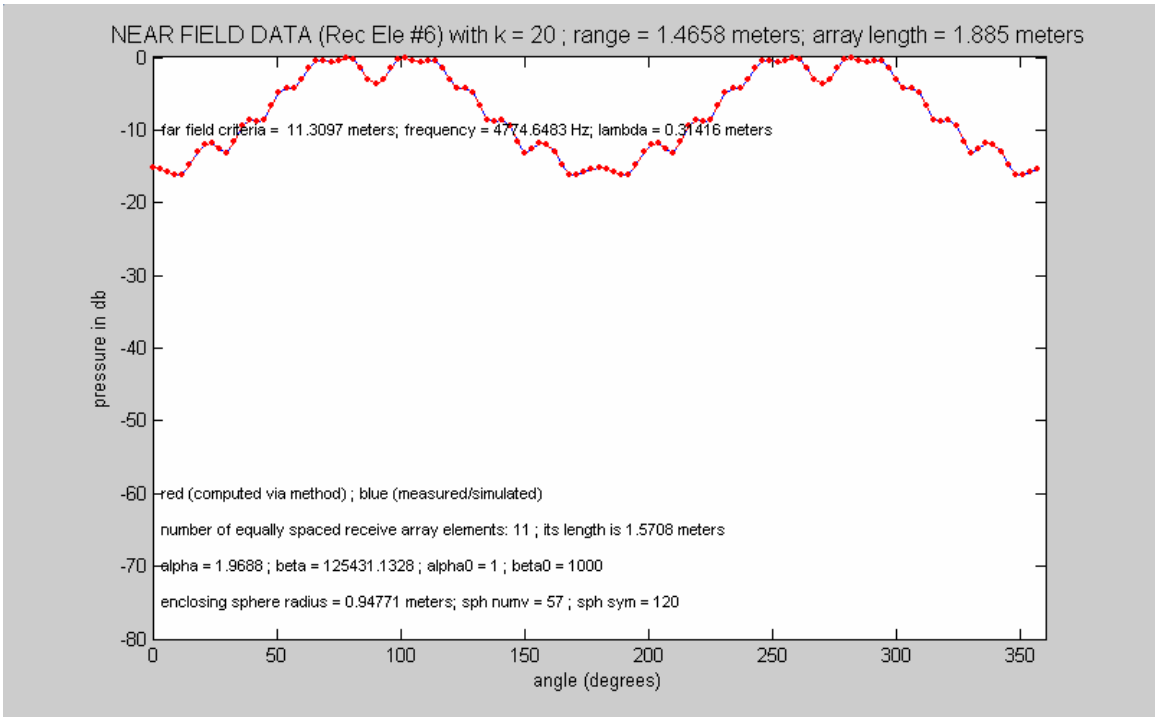


Figure 114 - Near Field Data – computed and simulated
Using 1.466 meter range, 1.571 meter long, 11 element near field array data

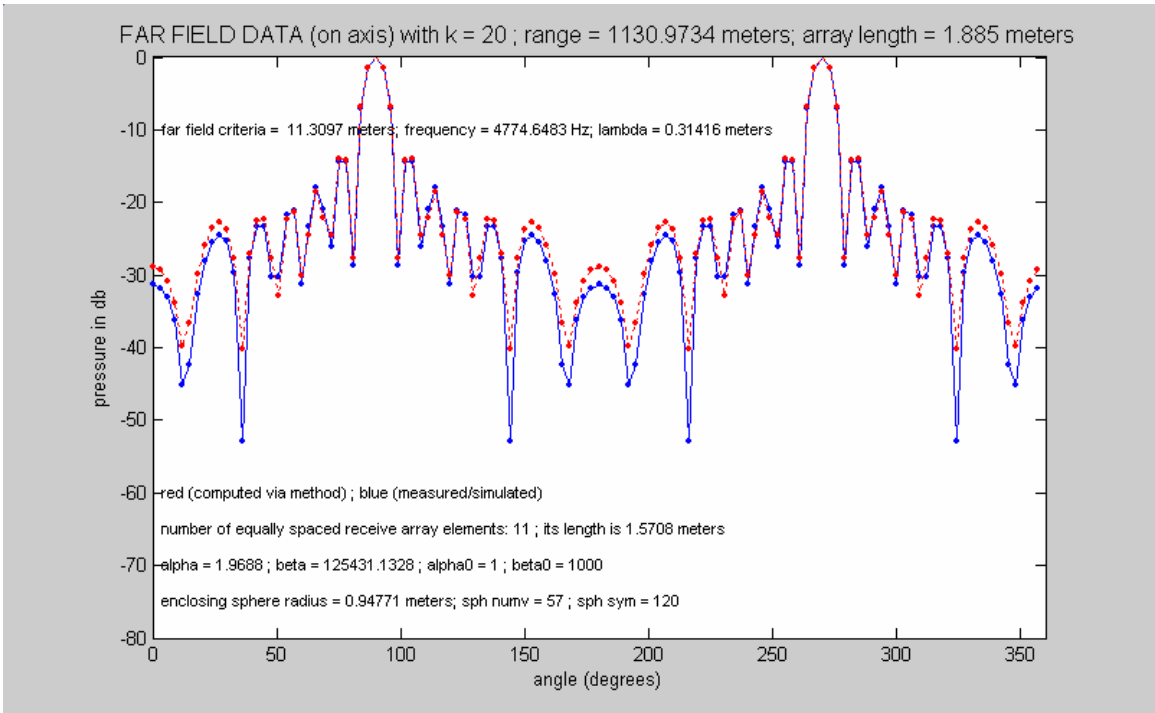


Figure 115 - Far Field Data – computed and simulated
Using 1.466 meter range, 1.571 meter long, 11 element near field array data

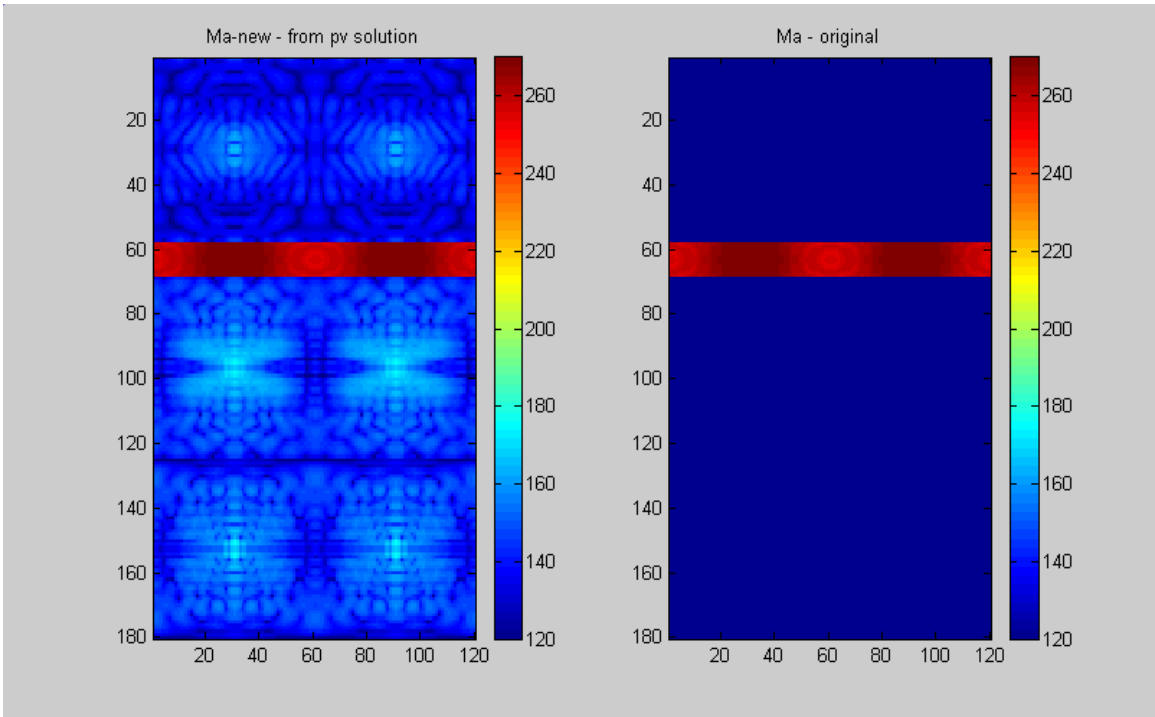


Figure 116 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
Using 1.466 meter range, 1.571 meter long, 11 element near field array data

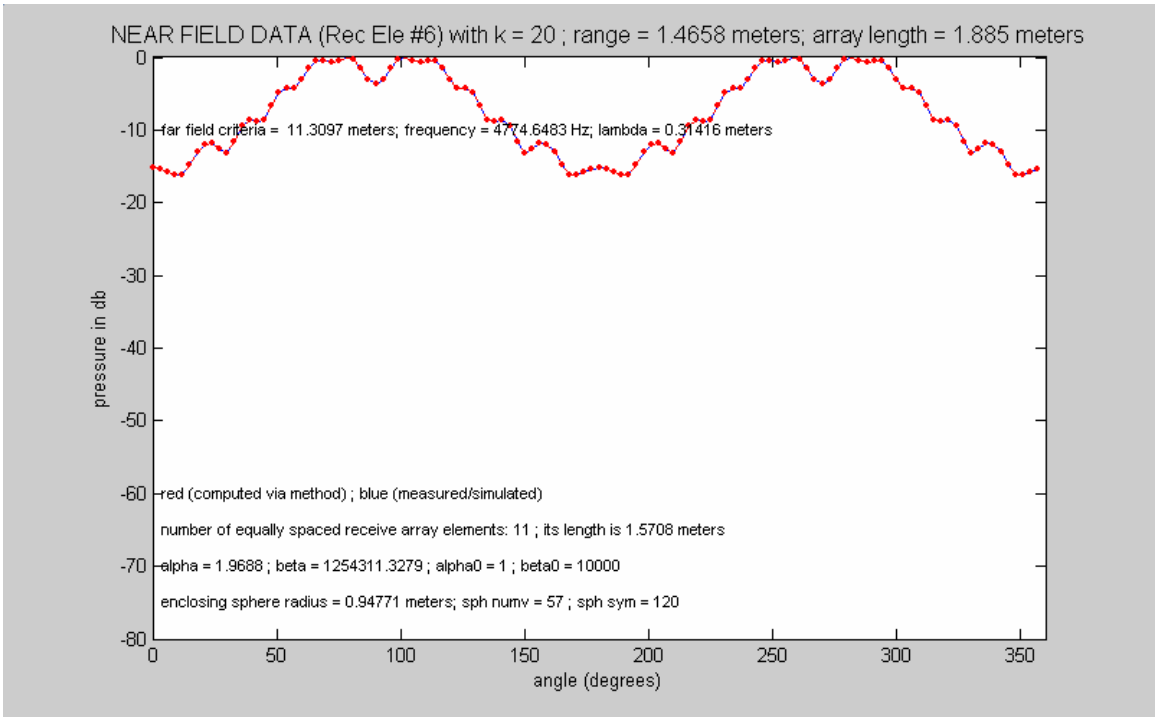


Figure 117 - Near Field Data – computed and simulated
Using 1.466 meter range, 1.571 meter long, 11 element near field array data

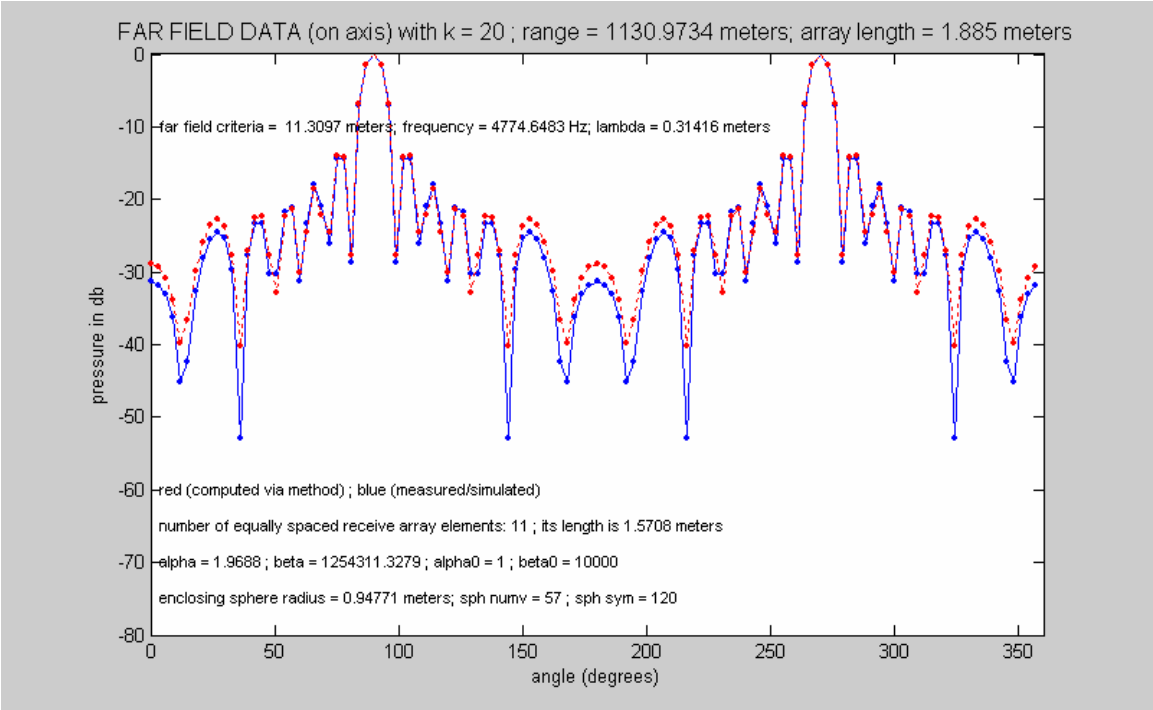


Figure 118 - Far Field Data – computed and simulated
Using 1.466 meter range, 1.571 meter long, 11 element near field array data

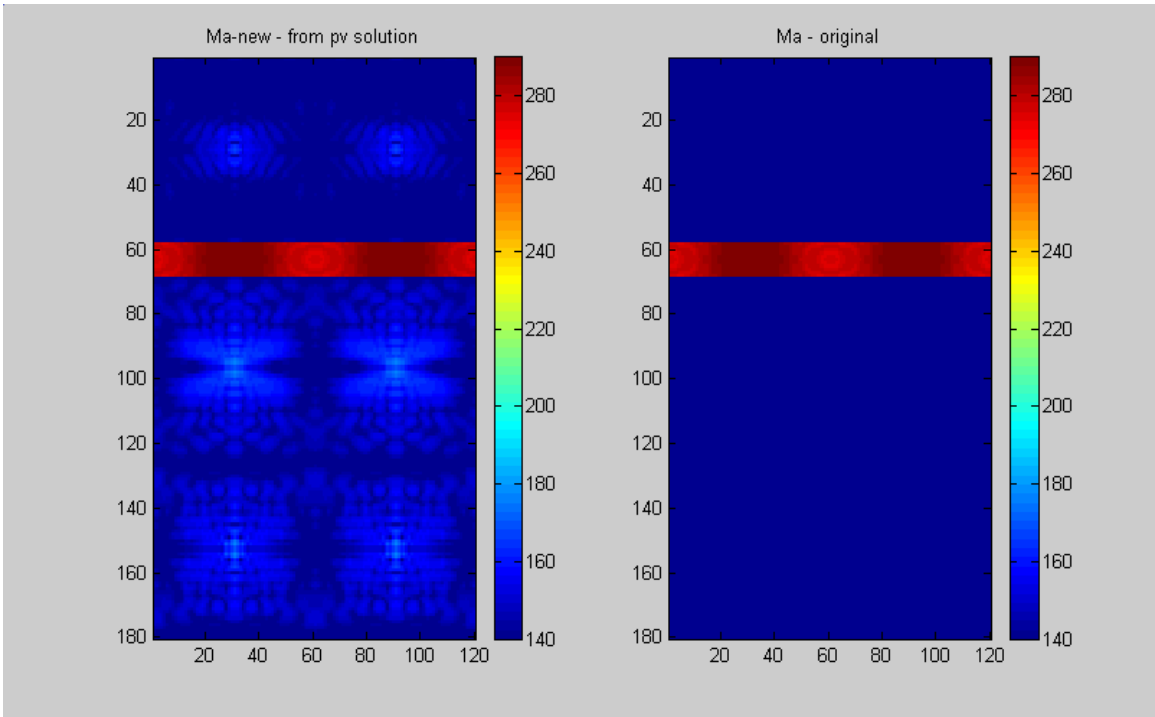


Figure 119 - $\mathbf{M_{a\text{-}new}}$ and $\mathbf{M_a}$ (original) Data Sets
Using 1.466 meter range, 1.571 meter long, 11 element near field array data

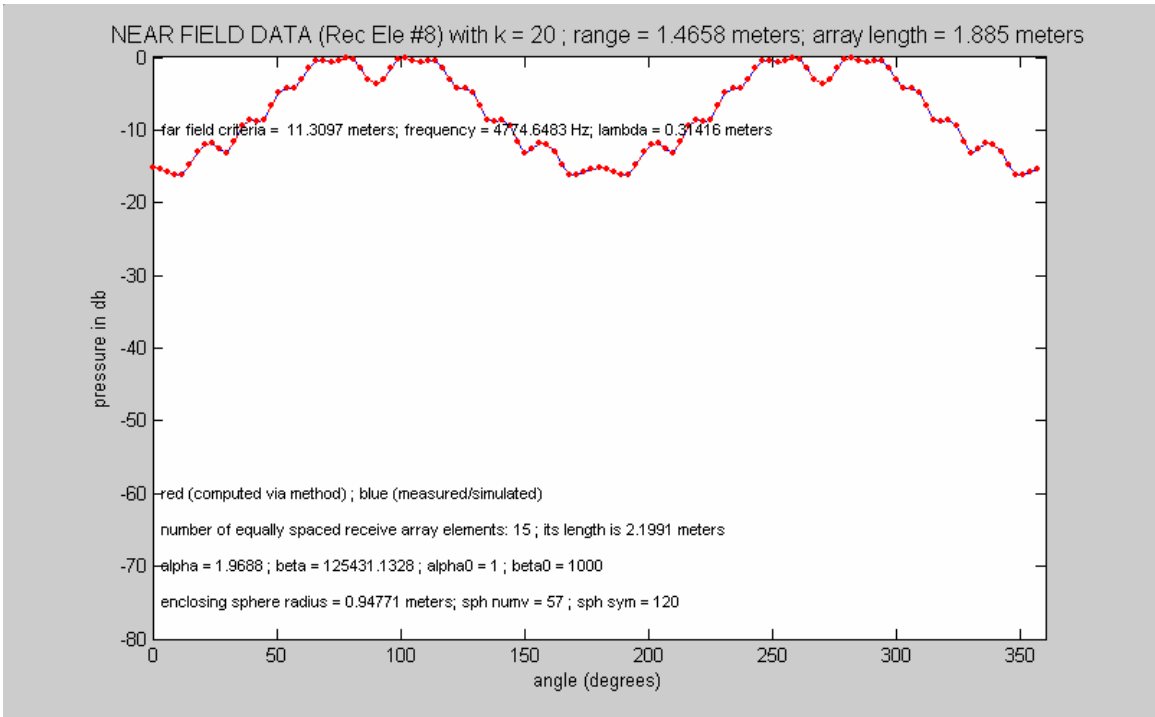


Figure 120 - Near Field Data – computed and simulated
Using 1.466 meter range, 2.199 meter long, 15 element near field array data

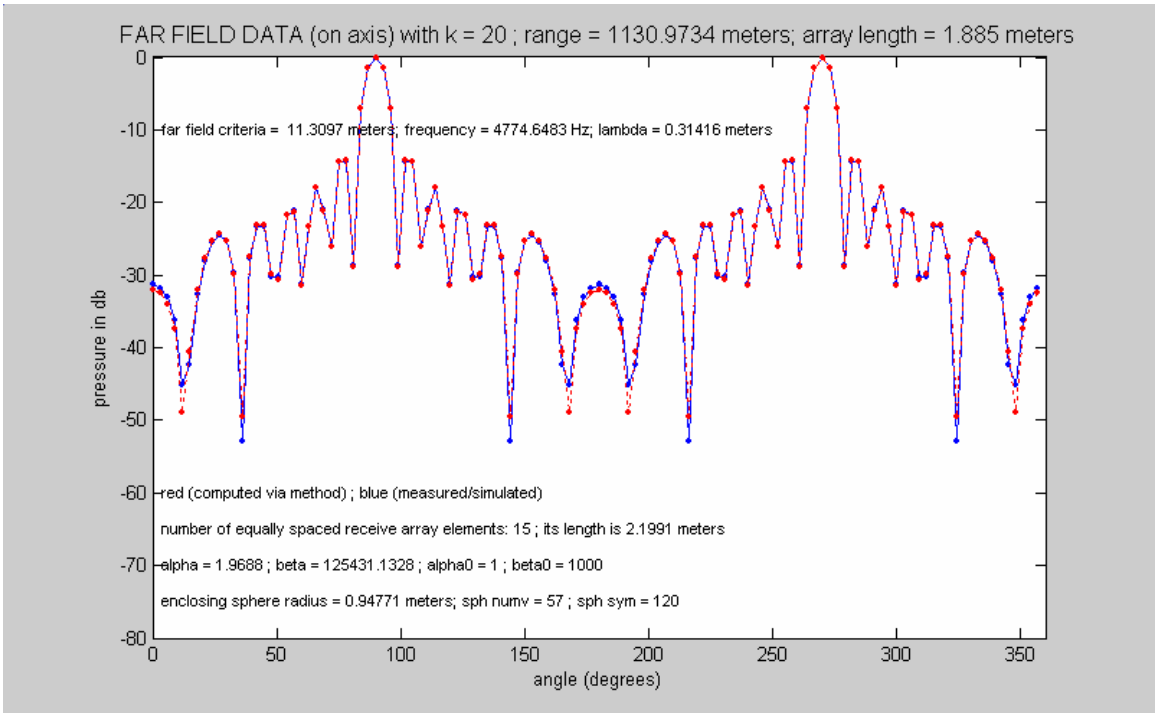


Figure 121 - Far Field Data – computed and simulated
Using 1.466 meter range, 2.199 meter long, 15 element near field array data

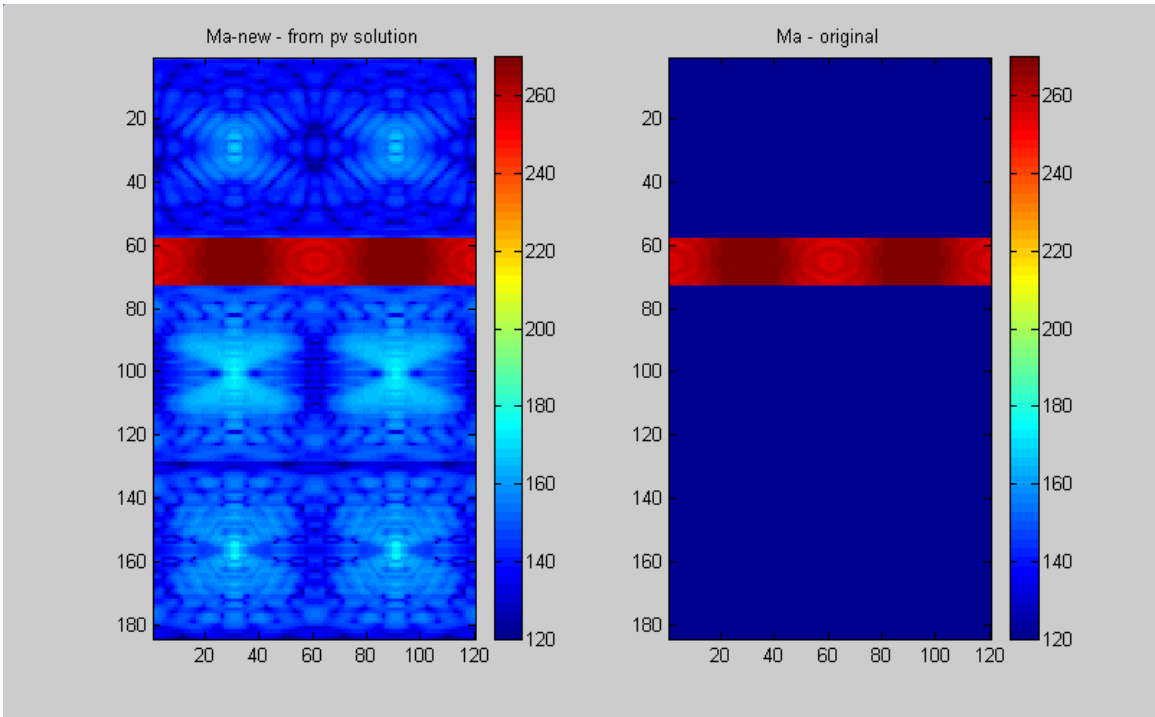


Figure 122 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
Using 1.466 meter range, 2.199 meter long, 15 element near field array data

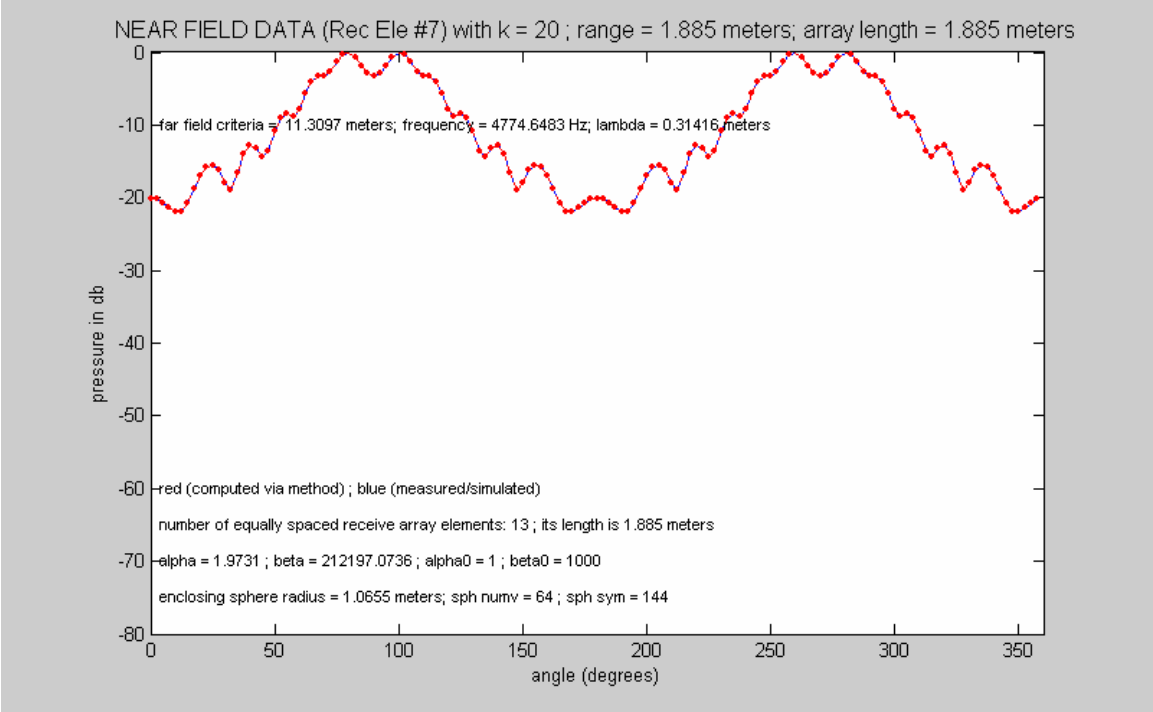


Figure 123 - Near Field Data – computed and simulated
Using 1.885 meter range, 1.885 meter long, 13 element near field array data

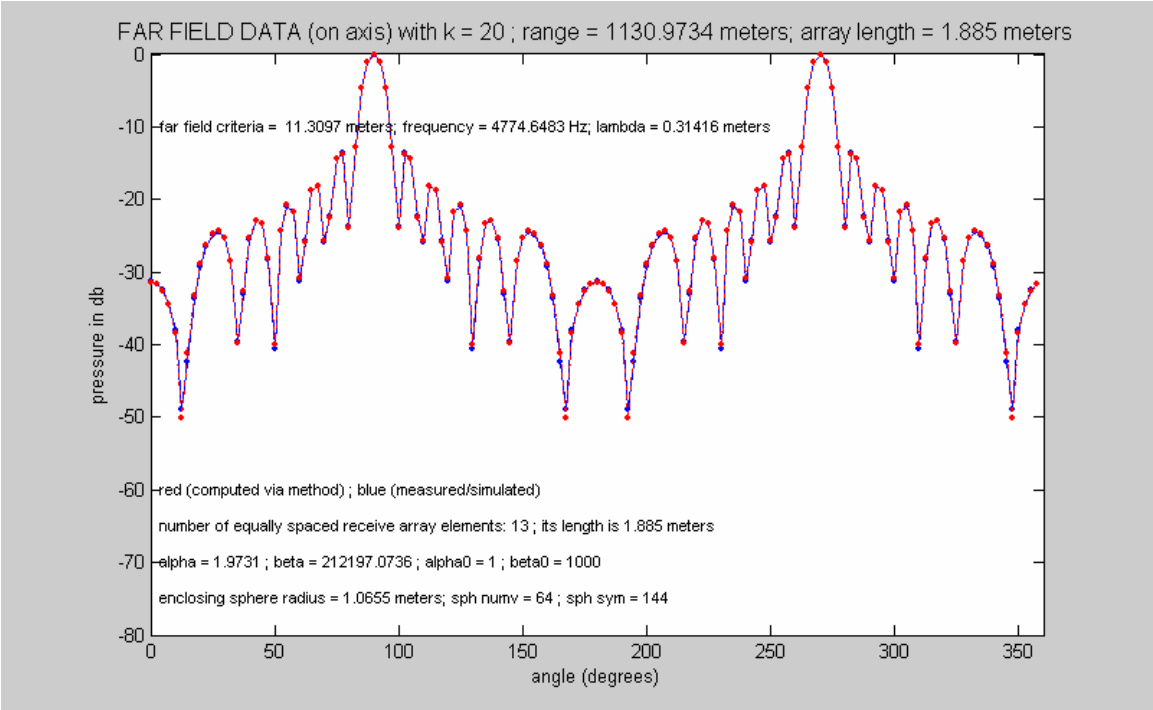


Figure 124 - Far Field Data – computed and simulated
Using 1.885 meter range, 1.885 meter long, 13 element near field array data

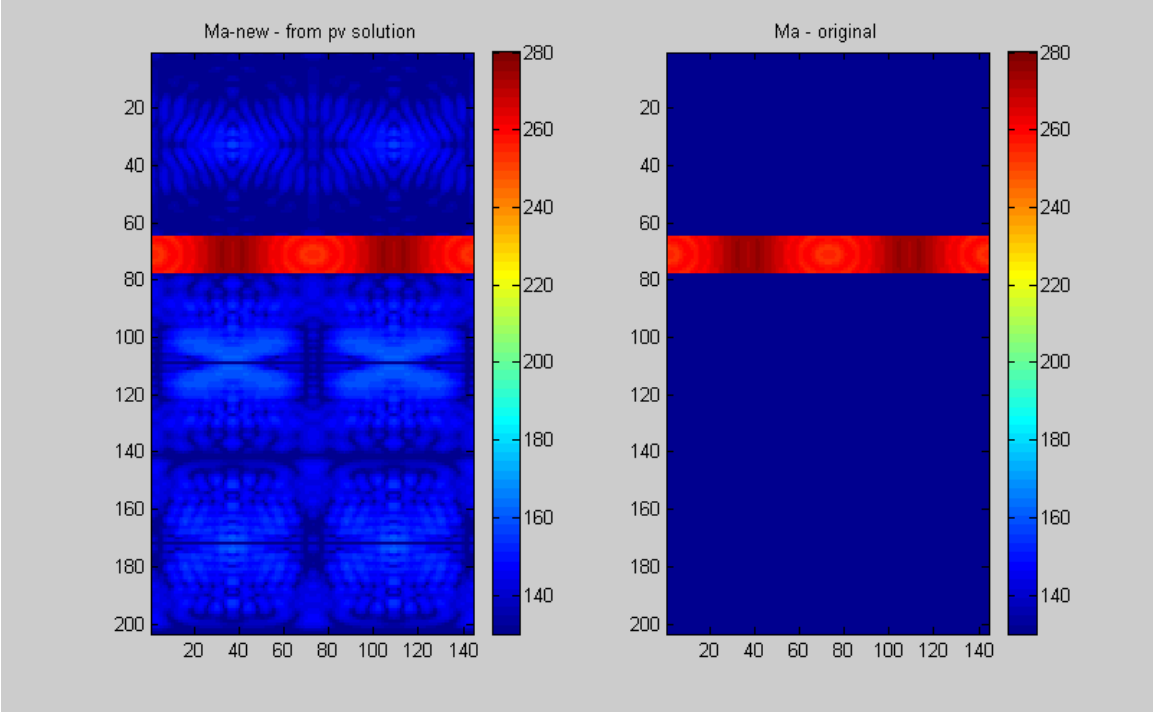


Figure 125 - $\mathbf{M}_{\mathbf{a\text{-}new}}$ and $\mathbf{M}_{\mathbf{a}}$ (original) Data Sets
Using 1.885 meter range, 1.885 meter long, 13 element near field array data

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A - EXAMPLE AB_MATRIX

The following pages show sections of a full augmented **AB_Matrix** as identified in this effort. This example is provided to assist in visualizing the matrix components, their relationship to one another and the overall circulant [7] nature of the matrix. The example provided details an enclosing sphere of very small order. Due to this small size, this matrix would not be suited for actual use but makes it ideal for instructional purposes.

For this example, an enclosing sphere was constructed with the following dimensions:

numv = 5	this represents the number of patches in the "north-south" direction
sph_sym = 10	this represents the number of "slices" that comprise the sphere
numh = 3	this represents the number of elements in the near field linear array

The enclosing sphere so obtained would look like the following:

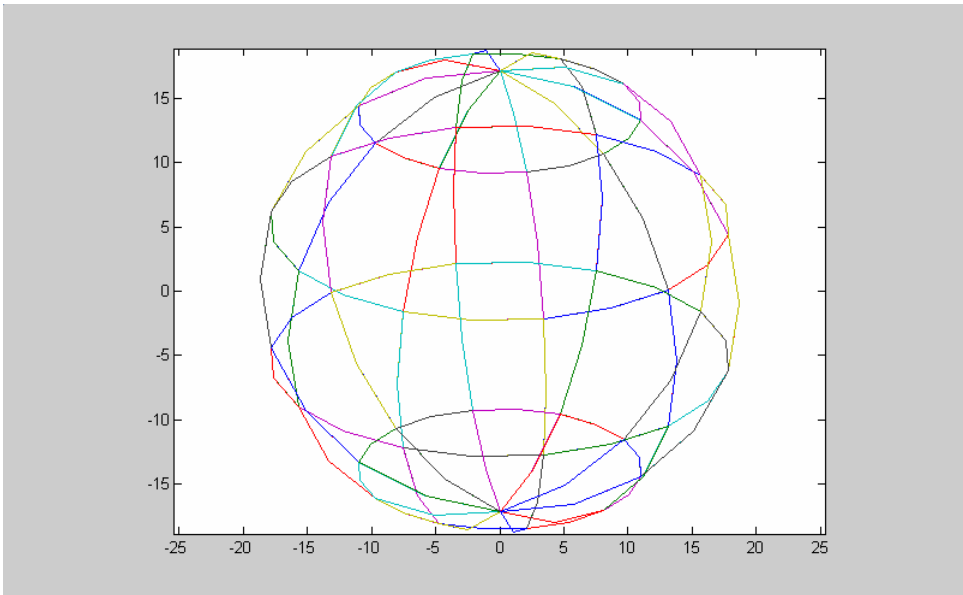


Figure A1 – CHIEF Enclosing Sphere

As discussed in the write-up, the constituents of the AB_Matrix are of the defined sizes listed below:

Matrix	**Size (rows x columns)**	**Example Matrix**
A	numv x (sph_numv * sph_sym)	5 x 50
B	" "	5 x 50
delG	numh x (sph_numv * sph_sym)	3 x 50
G	" "	3 x 50
PE	(sph_numv-1) x (sph_numv * sph_sym)	4 x 50
VE	" "	4 x 50

where:

PE = Pressure equivalence matrix
VE = Velocity equivalence matrix

The **AB_Matrix** is formed from the concatenation of the above components. This matrix takes the form of:

$$\begin{matrix} A_1 & B_1 & A_2 & B_2 & \dots & A_{50} & B_{50} \\ delG_1 & G_1 & delG_2 & G_2 & \dots & delG_{50} & G_{50} \\ PE_1 & 0 & PE_2 & 0 & & PE_{50} & 0 \\ 0 & VE_1 & 0 & VE_2 & & 0 & VE_{50} \end{matrix}$$

The size of one "slice" [2] of the **AB_matrix** is represented by the above arrangement. For the given example parameters, this matrix is of size 16 x 100. Since the full matrix is circulant due to the symmetry of the chosen enclosing surface (a sphere), the full **AB_matrix** is of size 160 x 100. (A matrix is considered to be circulant

if each of its rows, or columns, is a circular shift of the previous row, or column. If the matrix elements are considered to be matrices themselves and they obey the same circulant behavior as identified above, the matrix is more correctly identified as being block circulant.)

The pages that follow show some sections of the full example **AB_matrix**. It was not practical nor of any real value to include the entire 16,000 element matrix. Rather, some representative subset was provided. The matrix has been numbered for reference purposes with column numbers appearing across the top or bottom and row numbers appearing on the left hand side.

Since the array is circulant, the elements defined in the first slice (Rows 1 thru 16) and (Columns 1 thru 100) will repeat in a predictable manner throughout the rest of the matrix. Rows 1 thru 16 in Column 1 have been highlighted in red to show this repetition in subsequent slices. The data that appears in (Rows 1 thru 16), column 1 also appears in (Rows 17 thru 32), column11 and will appear in (Rows 33 thru 48), Column 21 and in other slices up until its final appearance in (Rows 145 thru 160), Column 91.

Examination of the other matrix elements shows that subsequent slices are created by shifting the present slice sph_sym (in our case this is 10) positions to the right and allowing those columns shifted past the last column (in our case this is 100) to be wrapped back to the beginning of the slice. This occurs a total of (sph_sym-1) times yielding the final matrix.

Another significant point in the construction of this matrix is the manner in which the adjacent patch pressure equivalence (PE) and velocity equivalence (VE) is represented. Along any single row of PE or VE, there will appear a single 1 value followed immediately by a single -1 value; all other elements in these rows are zero. This forms the conditions of $(p_i - p_{i+1})$ or $(v_i - v_{i+1})$ that is required for adjacent patch locations on the enclosing sphere. An examination of the matrix will actually show 1, 0, -1 (as in row 9, columns 1 thru 3) that is used to implement this condition. The zero in

this case is actually associated with the velocity term and thus does not interrupt the adjacency requirement between successive pressure terms. This is an artifact of the interleaving of the **A** and **B** matrices.

The **AB_matrix** represents the known terms in the overdetermined system that it creates. That is, in the example provided, there are 100 total unknowns. These correspond to 50 unknown pressures and 50 unknown velocities. A single to-be-solved-for pressure that will multiply all elements of column 1 will be generated via the least squares solution technique. A single to-be-solved- for velocity that will multiply all elements of column 2 will also be generated. Thus pressure values corresponding to all odd numbered columns and velocity values corresponding to all even numbered columns are ultimately generated.

Before leaving this appendix, it is significant to provide some additional comments on the symmetry requirement of the aforementioned components. In the MATLAB implementation, a single slice of the augmented **AB_matrix** is formed. Its symmetry conditions allow either the full **AB_matrix** to be determined (as was illustrated in this appendix) or it may be used directly with "rotational solve" tools [2] available in CHIEF for MATLAB to obtain a solution. This latter usage is preferred since it is computationally more efficient and imposes significantly less memory demands than the full defined **AB_matrix**.

To utilize these "rotational solve" tools, however, requires that the **AB_matrix** slice be representative of a section of the full block circulant **AB_matrix**. It is for this latter reason that such care was given to the definition of the enclosing sphere, the placement of the near field element points and the manner in which the adjacency conditions were defined. The symmetrical enclosing sphere satisfies the basic required circulant behavior since it is related to the field points in a very predictable manner. That is, a rotation of the enclosing sphere by one slice (with sph_sym being the total number of increments in a complete rotation) yields the same values of **A**, **B** , **G** and **delG** with

the exception that their placement in the matrix has shifted. In the example of this appendix, the columns of data shift 10 columns to the right for each incremental rotation.

Augmented AB_Matrix

	1 A and delG	2 B and G	3 A and delG	4 B and G	5 A and delG	6 B and G
1	**-2.995 + 0.17376i**	3.5684 + 0.92134i	-0.093182 + 0.15174i	0.54411 - 0.19015i	-0.013249 - 0.35889i	-0.032079 + 0.62409i
2	**-0.10782 + 0.0036907i**	0.52906 + 0.11537i	-2.6184 + 0.14508i	1.0352 - 0.82771i	-0.11272 + 0.20776i	0.7104 - 0.44097i
3	**0.017724 - 0.10239i**	-0.047656 + 0.21304i	-0.12407 + 0.16402i	0.69458 - 0.36421i	-2.6275 - 0.083952i	0.67693 + 0.59722i
4	**0.023804 + 0.097903i**	-0.01765 - 0.1466i	-0.02217 - 0.28557i	0.00057447 + 0.51634i	-0.11272 + 0.20776i	0.7104 - 0.44097i
5	**0.17511 + 0.27727i**	-0.17103 - 0.30889i	-0.10198 + 0.044271i	0.11917 - 0.098908i	-0.013249 - 0.35889i	-0.032079 + 0.62409i
6	**4.6858 + 2.4081i**	-10.891 - 14.991i	1.2014 - 6.6544i	-4.6272 + 28.753i	2.9742 - 6.3597i	-26.025 + 12.142i
7	**-3.0872 + 1.9549i**	17.786 + 3.1931i	4.5592 + 0.99307i	-29.777 - 6.7274i	6.4483 - 3.513i	-29.578 - 5.4269i
8	**-2.5756 - 0.091033i**	3.9504 + 17.378i	1.8408 + 1.0087i	-23.936 - 17.67i	2.9742 - 6.3597i	-26.025 + 12.142i
9	**1**	0	-1	0	0	0
10	**0**	0	1	0	-1	0
11	**0**	0	0	0	1	0
12	**0**	0	0	0	0	0
13	**0**	1	0	-1	0	0
14	**0**	0	0	1	0	-1
15	**0**	0	0	0	0	1
16	**0**	0	0	0	0	0
17	0.12274 - 0.024667i	-0.91857 - 0.71903i	0.12452 + 0.12967i	-0.15791 - 0.45878i	-0.25971 - 0.19699i	0.40766 + 0.35186i
18	-0.0027186 - 0.048651i	-0.039756 + 0.083106i	-0.1168 - 0.16394i	0.45809 + 0.83399i	0.05291 - 0.014402i	-0.1824 - 0.031035i
19	-0.028833 + 0.041861i	0.079527 - 0.06373i	0.13329 - 0.062714i	-0.34442 + 0.075509i	-0.11712 + 0.22948i	0.70567 - 0.51815i
20	0.14422 - 0.0026827i	-0.18576 - 0.021715i	-0.12969 + 0.0051617i	0.20041 - 0.0069979i	0.05291 - 0.014402i	-0.1824 - 0.031035i
21	0.25895 + 0.24221i	-0.26248 - 0.27254i	-0.033291 + 0.026667i	0.0081437 - 0.066227i	-0.25971 - 0.19699i	0.40766 + 0.35186i
22	-3.4698 + 0.78644i	8.1228 + 3.2625i	-7.4585 + 0.94841i	10.618 - 6.5525i	-18.52 + 17.577i	24.984 - 26.491i
23	0.38449 - 2.2907i	-5.7626 + 2.9136i	-3.2486 - 5.7139i	9.1501 + 9.1485i	-26.567 + 4.8747i	37.351 - 9.0999i
24	1.2807 - 1.1678i	-3.6595 - 2.6948i	-2.6672 - 5.0212i	6.7898 + 10.278i	-18.52 + 17.577i	24.984 - 26.491i
25	0	0	0	0	0	0
26	0	0	0	0	0	0
27	0	0	0	0	0	0
28	0	0	0	0	0	0
29	0	0	0	0	0	0
30	0	0	0	0	0	0
31	0	0	0	0	0	0
32	0	0	0	0	0	0

	7 A and delG	8 B and G	9 A and delG	10 B and G	11 A and delG	12 B and G
1	-0.10198 + 0.044271i	0.11917 - 0.098908i	0.17511 + 0.27727i	-0.17103 - 0.30889i	0.12274 - 0.024667i	-0.91857 - 0.71903i
2	-0.02217 - 0.28557i	0.00057447 + 0.51634i	0.023804 + 0.097903i	-0.01765 - 0.1466i	-0.0027186 - 0.048651i	-0.039756 + 0.083106i
3	-0.12407 + 0.16402i	0.69458 - 0.36421i	0.017724 - 0.10239i	-0.047656 + 0.21304i	-0.028833 + 0.041861i	0.079527 - 0.06373i
4	-2.6184 + 0.14508i	1.0352 - 0.82771i	-0.10782 + 0.0036907i	0.52906 + 0.11537i	0.14422 - 0.0026827i	-0.18576 - 0.021715i
5	-0.093182 + 0.15174i	0.54411 - 0.19015i	-2.995 + 0.17376i	3.5684 + 0.92134i	0.25895 + 0.24221i	-0.26248 - 0.27254i
6	1.8408 + 1.0087i	-23.936 - 17.67i	-2.5756 - 0.091033i	3.9504 + 17.378i	-1.4722 - 1.147i	0.75928 + 3.8456i
7	4.5592 + 0.99307i	-29.777 - 6.7274i	-3.0872 + 1.9549i	17.786 + 3.1931i	1.7204 - 0.40731i	-4.0252 - 2.8299i
8	1.2014 - 6.6544i	-4.6272 + 28.753i	4.6858 + 2.4081i	-10.891 - 14.991i	1.3635 + 0.95443i	1.3403 - 6.0311i
9	0	0	0	0	0	0
10	0	0	0	0	0	0
11	-1	0	0	0	0	0
12	1	0	-1	0	0	0
13	0	0	0	0	0	0
14	0	0	0	0	0	0
15	0	-1	0	0	0	0
16	0	1	0	-1	0	0
17	-0.033291 + 0.026667i	0.0081437 - 0.066227i	0.25895 + 0.24221i	-0.26248 - 0.27254i	**-2.995 + 0.17376i**	3.5684 + 0.92134i
18	-0.12969 + 0.0051617i	0.20041 - 0.0069979i	0.14422 - 0.0026827i	-0.18576 - 0.021715i	**-0.10782 + 0.0036907i**	0.52906 + 0.11537i
19	0.13329 - 0.062714i	-0.34442 + 0.075509i	-0.028833 + 0.041861i	0.079527 - 0.06373i	**0.017724 - 0.10239i**	-0.047656 + 0.21304i
20	-0.1168 - 0.16394i	0.45809 + 0.83399i	-0.0027186 - 0.048651i	-0.039756 + 0.083106i	**0.023804 + 0.097903i**	-0.01765 - 0.1466i
21	0.12452 + 0.12967i	-0.15791 - 0.45878i	0.12274 - 0.024667i	-0.91857 - 0.71903i	**0.17511 + 0.27727i**	-0.17103 - 0.30889i
22	-2.6672 - 5.0212i	6.7898 + 10.278i	1.2807 - 1.1678i	-3.6595 - 2.6948i	**4.6858 + 2.4081i**	-10.891 - 14.991i
23	-3.2486 - 5.7139i	9.1501 + 9.1485i	0.38449 - 2.2907i	-5.7626 + 2.9136i	**-3.0872 + 1.9549i**	17.786 + 3.1931i
24	-7.4585 + 0.94841i	10.618 - 6.5525i	-3.4698 + 0.78644i	8.1228 + 3.2625i	**-2.5756 - 0.091033i**	3.9504 + 17.378i
25	0	0	0	0	**1**	0
26	0	0	0	0	**0**	0
27	0	0	0	0	**0**	0
28	0	0	0	0	**0**	0
29	0	0	0	0	**0**	1
30	0	0	0	0	**0**	0
31	0	0	0	0	**0**	0
32	0	0	0	0	**0**	0

	19 **A and delG**	20 **B and G**	**21** **A and delG**	**22** **B and G**	**23** **A and delG**	**24** **B and G**
1	0.25895 + 0.24221i	-0.26248 - 0.27254i	-0.016032 - 0.014134i	0.044255 + 0.34347i	-0.081084 - 0.15998i	0.094108 + 0.39759i
2	0.14422 - 0.0026827i	-0.18576 - 0.021715i	0.10769 + 0.17937i	-0.16384 - 0.44722i	-0.054417 + 0.098339i	0.14141 - 0.23786i
3	-0.028833 + 0.041861i	0.079527 - 0.06373i	-0.013898 - 0.24811i	-0.016261 + 0.36104i	-0.074295 - 0.15654i	0.13711 + 0.22453i
4	-0.0027186 - 0.048651i	-0.039756 + 0.083106i	-0.17666 - 0.26745i	0.18622 + 0.32506i	0.16616 + 0.088226i	-0.18519 - 0.11633i
5	0.12274 - 0.024667i	-0.91857 - 0.71903i	0.41548 + 0.049814i	-0.43229 - 0.070006i	-0.12447 - 0.080101i	0.10459 + 0.12461i
6	1.3635 + 0.95443i	1.3403 - 6.0311i	0.77887 + 1.2599i	10.297 + 4.6152i	-16.185 - 2.3346i	-22.008 - 3.8157i
7	1.7204 - 0.40731i	-4.0252 - 2.8299i	-2.9368 + 0.28087i	-11.836 + 3.817i	1.0893 - 16.691i	3.0386 - 18.004i
8	-1.4722 - 1.147i	0.75928 + 3.8456i	-3.1626 - 3.7335i	-9.4768 - 9.6625i	-1.4769 - 17.695i	-2.173 - 14.666i
9	0	0	0	0	0	0
10	0	0	0	0	0	0
11	0	0	0	0	0	0
12	0	0	0	0	0	0
13	0	0	0	0	0	0
14	0	0	0	0	0	0
15	0	0	0	0	0	0
16	0	0	0	0	0	0
17	0.17511 + 0.27727i	-0.17103 - 0.30889i	0.12274 - 0.024667i	-0.91857 - 0.71903i	0.12452 + 0.12967i	-0.15791 - 0.45878i
18	0.023804 + 0.097903i	-0.01765 - 0.1466i	-0.0027186 - 0.048651i	-0.039756 + 0.083106i	-0.1168 - 0.16394i	0.45809 + 0.83399i
19	0.017724 - 0.10239i	-0.047656 + 0.21304i	-0.028833 + 0.041861i	0.079527 - 0.06373i	0.13329 - 0.062714i	-0.34442 + 0.075509i
20	-0.10782 + 0.0036907i	0.52906 + 0.11537i	0.14422 - 0.0026827i	-0.18576 - 0.021715i	-0.12969 + 0.0051617i	0.20041 - 0.0069979i
21	-2.995 + 0.17376i	3.5684 + 0.92134i	0.25895 + 0.24221i	-0.26248 - 0.27254i	-0.033291 + 0.026667i	0.0081437 - 0.066227i
22	-2.5756 - 0.091033i	3.9504 + 17.378i	-1.4722 - 1.147i	0.75928 + 3.8456i	2.8948 - 3.3181i	3.8416 - 13.27i
23	-3.0872 + 1.9549i	17.786 + 3.1931i	1.7204 - 0.40731i	-4.0252 - 2.8299i	5.1313 + 2.2799i	16.787 + 3.8307i
24	4.6858 + 2.4081i	-10.891 - 14.991i	1.3635 + 0.95443i	1.3403 - 6.0311i	7.4033 + 3.3046i	19.042 + 9.3051i
25	0	0	0	0	0	0
26	0	0	0	0	0	0
27	0	0	0	0	0	0
28	-1	0	0	0	0	0
29	0	0	0	0	0	0
30	0	0	0	0	0	0
31	0	0	0	0	0	0
32	0	-1	0	0	0	0

127	0	0	0	0	0	0
128	0	-1	0	0	0	0
129	0.25895 + 0.24221i	-0.26248 - 0.27254i	**-2.995 + 0.17376i**	3.5684 + 0.92134i	-0.093182 + 0.15174i	0.54411 - 0.19015i
130	0.14422 - 0.0026827i	-0.18576 - 0.021715i	**-0.10782 + 0.0036907i**	0.52906 + 0.11537i	-2.6184 + 0.14508i	1.0352 - 0.82771i
131	-0.028833 + 0.041861i	0.079527 - 0.06373i	**0.017724 - 0.10239i**	-0.047656 + 0.21304i	-0.12407 + 0.16402i	0.69458 - 0.36421i
132	-0.0027186 - 0.048651i	-0.039756 + 0.083106i	**0.023804 + 0.097903i**	-0.01765 - 0.1466i	-0.02217 - 0.28557i	0.00057447 + 0.51634i
133	0.12274 - 0.024667i	-0.91857 - 0.71903i	**0.17511 + 0.27727i**	-0.17103 - 0.30889i	-0.10198 + 0.044271i	0.11917 - 0.098908i
134	1.2807 - 1.1678i	-3.6595 - 2.6948i	**4.6858 + 2.4081i**	-10.891 - 14.991i	1.2014 - 6.6544i	-4.6272 + 28.753i
135	0.38449 - 2.2907i	-5.7626 + 2.9136i	**-3.0872 + 1.9549i**	17.786 + 3.1931i	4.5592 + 0.99307i	-29.777 - 6.7274i
136	-3.4698 + 0.78644i	8.1228 + 3.2625i	**-2.5756 - 0.091033i**	3.9504 + 17.378i	1.8408 + 1.0087i	-23.936 - 17.67i
137	0	0	**1**	0	-1	0
138	0	0	**0**	0	1	0
139	0	0	**0**	0	0	0
140	0	0	**0**	0	0	0
141	0	0	**0**	1	0	-1
142	0	0	**0**	0	0	1
143	0	0	**0**	0	0	0
144	0	0	**0**	0	0	0
145	0.41548 + 0.049814i	-0.43229 - 0.070006i	0.12274 - 0.024667i	-0.91857 - 0.71903i	0.12452 + 0.12967i	-0.15791 - 0.45878i
146	-0.17666 - 0.26745i	0.18622 + 0.32506i	-0.0027186 - 0.048651i	-0.039756 + 0.083106i	-0.1168 - 0.16394i	0.45809 + 0.83399i
147	-0.013898 - 0.24811i	-0.016261 + 0.36104i	-0.028833 + 0.041861i	0.079527 - 0.06373i	0.13329 - 0.062714i	-0.34442 + 0.075509i
148	0.10769 + 0.17937i	-0.16384 - 0.44722i	0.14422 - 0.0026827i	-0.18576 - 0.021715i	-0.12969 + 0.0051617i	0.20041 - 0.0069979i
149	-0.016032 - 0.014134i	0.044255 + 0.34347i	0.25895 + 0.24221i	-0.26248 - 0.27254i	-0.033291 + 0.026667i	0.0081437 - 0.066227i
150	-2.8241 + 4.7513i	6.5666 - 10.689i	-3.4698 + 0.78644i	8.1228 + 3.2625i	-7.4585 + 0.94841i	10.618 - 6.5525i
151	2.6929 + 6.5929i	-5.0602 - 11.867i	0.38449 - 2.2907i	-5.7626 + 2.9136i	-3.2486 - 5.7139i	9.1501 + 9.1485i
152	5.4551 - 6.5507i	-8.0569 + 10.137i	1.2807 - 1.1678i	-3.6595 - 2.6948i	-2.6672 - 5.0212i	6.7898 + 10.278i
153	0	0	0	0	0	0
154	0	0	0	0	0	0
155	0	0	0	0	0	0
156	0	0	0	0	0	0
157	0	0	0	0	0	0
158	0	0	0	0	0	0
159	0	0	0	0	0	0
160	0	0	0	0	0	0
	79	80	**81**	**82**	**83**	**84**
	A and delG	**B and G**	**A and delG**	**B and G**	**A and delG**	**B and G**

127	0	0	0	0	0	0
128	0	0	0	0	0	0
129	0.12274 - 0.024667i	-0.91857 - 0.71903i	0.12452 + 0.12967i	-0.15791 - 0.45878i	-0.25971 - 0.19699i	0.40766 + 0.35186i
130	-0.0027186 - 0.048651i	-0.039756 + 0.083106i	-0.1168 - 0.16394i	0.45809 + 0.83399i	0.05291 - 0.014402i	-0.1824 - 0.031035i
131	-0.028833 + 0.041861i	0.079527 - 0.06373i	0.13329 - 0.062714i	-0.34442 + 0.075509i	-0.11712 + 0.22948i	0.70567 - 0.51815i
132	0.14422 - 0.0026827i	-0.18576 - 0.021715i	-0.12969 + 0.0051617i	0.20041 - 0.0069979i	0.05291 - 0.014402i	-0.1824 - 0.031035i
133	0.25895 + 0.24221i	-0.26248 - 0.27254i	-0.033291 + 0.026667i	0.0081437 - 0.066227i	-0.25971 - 0.19699i	0.40766 + 0.35186i
134	-1.4722 - 1.147i	0.75928 + 3.8456i	2.8948 - 3.3181i	3.8416 - 13.27i	7.2473 + 21.658i	18.329 + 37.397i
135	1.7204 - 0.40731i	-4.0252 - 2.8299i	5.1313 + 2.2799i	16.787 + 3.8307i	-8.2609 + 23.111i	-9.7121 + 43.426i
136	1.3635 + 0.95443i	1.3403 - 6.0311i	7.4033 + 3.3046i	19.042 + 9.3051i	7.2473 + 21.658i	18.329 + 37.397i
137	0	0	0	0	0	0
138	0	0	0	0	0	0
139	0	0	0	0	0	0
140	0	0	0	0	0	0
141	0	0	0	0	0	0
142	0	0	0	0	0	0
143	0	0	0	0	0	0
144	0	0	0	0	0	0
145	**-2.995 + 0.17376i**	3.5684 + 0.92134i	-0.093182 + 0.15174i	0.54411 - 0.19015i	-0.013249 - 0.35889i	-0.032079 + 0.62409i
146	**-0.10782 + 0.0036907i**	0.52906 + 0.11537i	-2.6184 + 0.14508i	1.0352 - 0.82771i	-0.11272 + 0.20776i	0.7104 - 0.44097i
147	**0.017724 - 0.10239i**	-0.047656 + 0.21304i	-0.12407 + 0.16402i	0.69458 - 0.36421i	-2.6275 - 0.083952i	0.67693 + 0.59722i
148	**0.023804 + 0.097903i**	-0.01765 - 0.1466i	-0.02217 - 0.28557i	0.00057447 + 0.51634i	-0.11272 + 0.20776i	0.7104 - 0.44097i
149	**0.17511 + 0.27727i**	-0.17103 - 0.30889i	-0.10198 + 0.044271i	0.11917 - 0.098908i	-0.013249 - 0.35889i	-0.032079 + 0.62409i
150	**4.6858 + 2.4081i**	-10.891 - 14.991i	1.2014 - 6.6544i	-4.6272 + 28.753i	2.9742 - 6.3597i	-26.025 + 12.142i
151	**-3.0872 + 1.9549i**	17.786 + 3.1931i	4.5592 + 0.99307i	-29.777 - 6.7274i	6.4483 - 3.513i	-29.578 - 5.4269i
152	**-2.5756 - 0.091033i**	3.9504 + 17.378i	1.8408 + 1.0087i	-23.936 - 17.67i	2.9742 - 6.3597i	-26.025 + 12.142i
153	**1**	0	-1	0	0	0
154	**0**	0	1	0	-1	0
155	**0**	0	0	0	1	0
156	**0**	0	0	0	0	0
157	**0**	1	0	-1	0	0
158	**0**	0	0	1	0	-1
159	**0**	0	0	0	0	1
160	**0**	0	0	0	0	0
	91	92	93	94	95	96
	A and delG	**B and G**	**A and delG**	**B and G**	**A and delG**	**B and G**

127	0	0	0	0
128	0	0	0	0
129	-0.033291 + 0.026667i	0.0081437 - 0.066227i	0.25895 + 0.24221i	-0.26248 - 0.27254i
130	-0.12969 + 0.0051617i	0.20041 - 0.0069979i	0.14422 - 0.0026827i	-0.18576 - 0.021715i
131	0.13329 - 0.062714i	-0.34442 + 0.075509i	-0.028833 + 0.041861i	0.079527 - 0.06373i
132	-0.1168 - 0.16394i	0.45809 + 0.83399i	-0.0027186 - 0.048651i	-0.039756 + 0.083106i
133	0.12452 + 0.12967i	-0.15791 - 0.45878i	0.12274 - 0.024667i	-0.91857 - 0.71903i
134	7.4033 + 3.3046i	19.042 + 9.3051i	1.3635 + 0.95443i	1.3403 - 6.0311i
135	5.1313 + 2.2799i	16.787 + 3.8307i	1.7204 - 0.40731i	-4.0252 - 2.8299i
136	2.8948 - 3.3181i	3.8416 - 13.27i	-1.4722 - 1.147i	0.75928 + 3.8456i
137	0	0	0	0
138	0	0	0	0
139	0	0	0	0
140	0	0	0	0
141	0	0	0	0
142	0	0	0	0
143	0	0	0	0
144	0	0	0	0
145	-0.10198 + 0.044271i	0.11917 - 0.098908i	0.17511 + 0.27727i	-0.17103 - 0.30889i
146	-0.02217 - 0.28557i	0.00057447 + 0.51634i	0.023804 + 0.097903i	-0.01765 - 0.1466i
147	-0.12407 + 0.16402i	0.69458 - 0.36421i	0.017724 - 0.10239i	-0.047656 + 0.21304i
148	-2.6184 + 0.14508i	1.0352 - 0.82771i	-0.10782 + 0.0036907i	0.52906 + 0.11537i
149	-0.093182 + 0.15174i	0.54411 - 0.19015i	-2.995 + 0.17376i	3.5684 + 0.92134i
150	1.8408 + 1.0087i	-23.936 - 17.67i	-2.5756 - 0.091033i	3.9504 + 17.378i
151	4.5592 + 0.99307i	-29.777 - 6.7274i	-3.0872 + 1.9549i	17.786 + 3.1931i
152	1.2014 - 6.6544i	-4.6272 + 28.753i	4.6858 + 2.4081i	-10.891 - 14.991i
153	0	0	0	0
154	0	0	0	0
155	-1	0	0	0
156	1	0	-1	0
157	0	0	0	0
158	0	0	0	0
159	0	-1	0	0
160	0	1	0	-1
	97	98	99	100
	A and delG	**B and G**	**A and delG**	**B and G**

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B – LEAST SQUARES SOLUTION TECHNIQUE

The least squares technique allows an overdetermined system ("m" equations in "n" unknowns, with m>n) to be evaluated in an optimized manner. This optimized manner results in values of **x** that minimize the quantity $|\mathbf{Ax} - \mathbf{b}|^2$ for the originating equation of **Ax = b**.

The vector **x** that minimizes $|\mathbf{Ax} - \mathbf{b}|^2$ is the solution to the normal equations:

$$\mathbf{A}^T \mathbf{Ax} = \mathbf{A}^T \mathbf{b}$$

This vector $\mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}$ is the least squares solution to **Ax =b** [7].

This appendix seeks to provide the reader with a little more understanding of the mechanism of this technique. A relatively simple example of a sinusoidal wave corrupted with a higher frequency random noise is generated. The least squares solution technique is then used to determine a best fit "pure" sine wave to the input data provided. The sine wave so obtained minimizes the error per the aforementioned solution definitions.

The least squares solution technique can best be envisioned as a means to provide a "best" solution to a long list of equations. Each equation has the same number and type of unknown constants. For the example chosen, a sine wave can be written in terms of both sine and cosine components. That is, this signal can be written as the sum of these two entities via:

$$D \sin(\omega t) + E \cos(\omega t) \qquad (1)$$

where:

ω = radian frequency

t = time

D and E are some defined constants

In addition, the signal may have a DC component, so Equation (1) may be augmented to:

$$D \sin(\omega t) + E \cos(\omega t) + F \qquad (2)$$

where:

F = some constant DC level

Equation (2) can uniquely specify a given radian frequency sinusoidally varying signal by defining the values for the constants D, E and F. Phase delays can be introduced by the proper weighting of the D and E values. That is, if E=0, then the signal so defined has pure sine characteristics. If D =0, then the signal so defined has pure cosine characteristics. Non-zero values for both D and E yield sinusoidally varying signals with some defined phase shift.

Under the assumptions that the radian frequency, ω, is a known value and that time, t, can be incrementally adjusted (t_1, t_2, t_3, t_4, etc), a set of equations defining the signal may be created.

$$D \sin(\omega t_1) + E \cos(\omega t_1) + F \qquad (3)$$
$$D \sin(\omega t_2) + E \cos(\omega t_2) + F$$
$$D \sin(\omega t_3) + E \cos(\omega t_3) + F$$
$$D \sin(\omega t_4) + E \cos(\omega t_4) + F$$
$$.$$
$$.$$
$$.$$

The equations defined above consist of 3 unknowns (D, E and F) that will eventually need to be solved for. To assist in this solution, the equations specified in (3) above are equated to a given a set of measured data values, **m**. This leads to:

$$D \sin(\omega t_1) + E \cos(\omega t_1) + F = m_1 \quad (4)$$
$$D \sin(\omega t_2) + E \cos(\omega t_2) + F = m_2$$
$$D \sin(\omega t_3) + E \cos(\omega t_3) + F = m_3$$
$$D \sin(\omega t_4) + E \cos(\omega t_4) + F = m_4$$
.
.
.

The expressions above form the overdetermined set of equations that will be solved via least squares solution technique. The equations in (4) are rewritten in the form of matrices yielding:

$$\begin{matrix} \sin(\omega t_1) & \cos(\omega t_1) & 1 \\ \sin(\omega t_2) & \cos(\omega t_2) & 1 \\ \sin(\omega t_3) & \cos(\omega t_3) & 1 \\ \sin(\omega t_4) & \cos(\omega t_4) & 1 \end{matrix} * [D\ E\ F]' = \begin{matrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{matrix} \quad (5)$$

.
.
.

where D multiplies all values in column 1 , E multiplies all values in column 2 and F multiplies all values in column 3 of the first matrix.

The matrices in Equation (5) are now in the form of the defining equation **Ax =b** with:

$$A = \begin{matrix} \sin(\omega t_1) & \cos(\omega t_1) & 1 \\ \sin(\omega t_2) & \cos(\omega t_2) & 1 \\ \sin(\omega t_3) & \cos(\omega t_3) & 1 \\ \sin(\omega t_4) & \cos(\omega t_4) & 1 \\ . & & \\ . & & \\ . & & \end{matrix}$$

$$b = \begin{matrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ . \\ . \\ . \end{matrix}$$

and

$$x = [\, D \; E \; F]'$$

The solution may now proceed via the least squares solution technique detailed at the beginning of this appendix.

For the example chosen, a 1 Hz sinusoidally varying signal is sampled at every 1/256 seconds. The data corresponding to the first 10 rows (of the 1024 total) of the **A** matrix above was computed to be:

A

1	0.02454	0.99970	1
2	0.04907	0.99880	1
3	0.07357	0.99729	1
4	0.09802	0.99518	1
5	0.12241	0.99248	1
6	0.14673	0.98918	1
7	0.17096	0.98528	1
8	0.19509	0.98079	1
9	0.21910	0.97570	1
10	0.24298	0.97003	1

The data corresponding to the first 10 rows of the b matrix were determined to be:

b

1	0.68685
2	0.48398
3	0.64596
4	0.64235
5	0.81405
6	0.80746
7	0.74193
8	0.63214
9	0.93579
10	0.84604

In like fashion as matrix **A**, vector **b** is composed of 1024 total rows. The data presented in vector **b** consists of the sinusoidally varying signal corrupted by some higher frequency noise. This corruption by noise explains why the values in b do not vary in a purely sinusoidal manner.

The solution, x, for this set-up was determined to be:

	x
D	1.50520
E	0.00903
F	0.50071

Recall that D represents the sine component, E represents the cosine component and F represents the DC offset of the measured signal.

The chart that follows shows the original signal (blue crosses) along with the least squares solution for a best fit sinusoidally varying signal (red line).

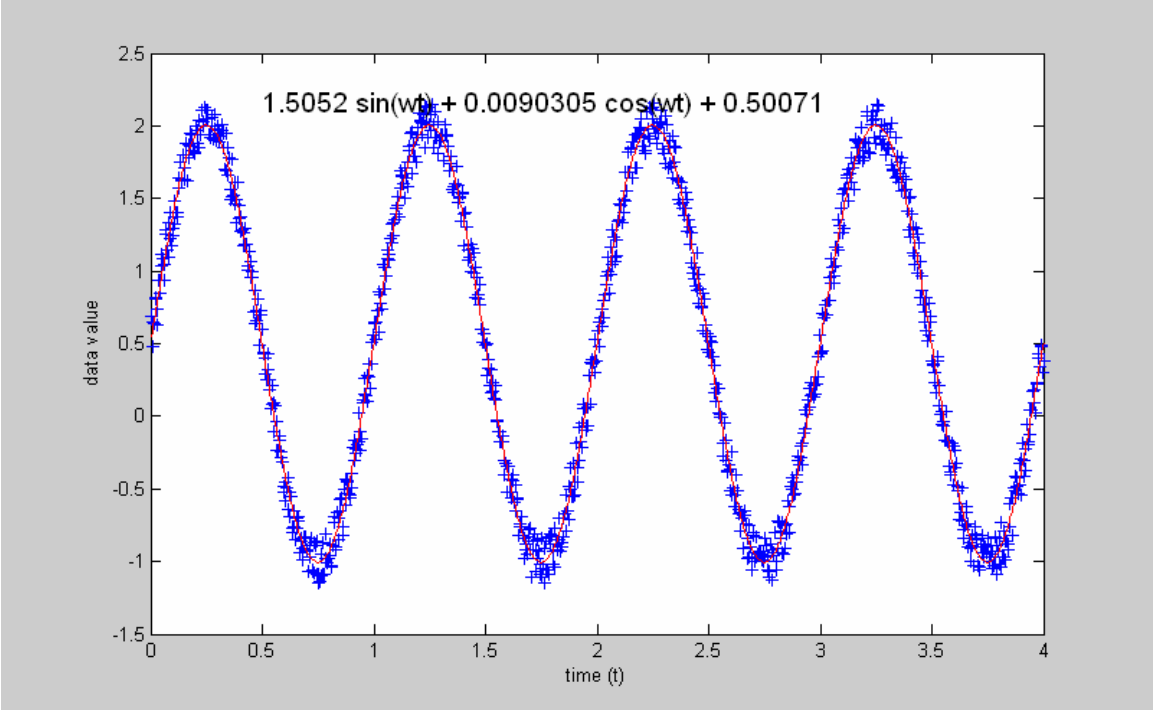


Figure B1 – Least Squares Solution for Sinewave with Noise and DC Offset

It is significant here to address the requirement to have an over determined system to utilize the method of least squares. The sinusoidal signal discussed in this section was

represented by a system of 1024 equations in 3 unknowns. This led, quite obviously, to an over determined system (since m=1024 was much greater than n =3).

The thesis method required the incorporation of theoretical and physical parameters to achieve an over determined status. The example afforded by the **AB_Matrix** construction of Appendix A can best illustrate this.

Appendix A described an **AB_Matrix** with the following constituents:

numv = 5 ; this represents the number of patches in the "north-south" direction
sph_sym = 10 ; this represents the number of "slices" that comprise the sphere
numh = 3 ; this represents the number of elements in the near field linear array

The corresponding system required that a solution be determined for 50 individual patches (numv x sph_sym). Each patch required a pressure and velocity solution; this, therefore, lead to the need to solve for a total of 100 unknowns (50 pressures and 50 velocities).

1) The use of the first Helmholtz relation, 0 =**-Ap+Bv**, provides a total of 50 equations that define the physical system. The 50 equations result from the fact that the **AB_Matrix** is circulant and is based upon the number of vertical patches (numv) and the symmetry order (sph_sym) of the enclosing sphere. These 50 equations are, however, insufficient for solution via least squares since 100 unknowns must be solved for. This system incorporating just this Helmholtz relation is, therefore, under determined.

2) The use of near field data was essential to the overall effort since physical characteristics of the system needed to be quantified. The incorporation of these physical characteristics was embodied in the second Helmholtz relation, pressure at a field point = **delG p + G v**. The measured near field element

data with the above relationship formed another set of defining equations for the system.

The equations formed from this relationship corresponded to the number of elements in the near field array (numh =3) and the symmetry order (sph_sym=10). Thus, for the example, 30 equations were associated with this Helmholtz relation.

These 30 equations are added to the initial 50 equations from the previous step yielding 80 total equations defining the system. These 80 equations, however, still can not solve for the 100 unknowns; the system is still under determined.

In this example, a near field array of 3 elements was selected. It would have been possible to define a near field array with more elements that could have met the over determined requirement of the least squares methodology (when used with the previous Helmholtz relations) . For example, a 10 element array would have yielded 100 equations. These 100 plus the previous 50 would have yielded a sufficient number for a least squares solution.

For this relatively simple example, a 10 element array would not have presented any real difficulties to implement. For other systems, the number of near field elements to meet the over determined requirement could approach staggering proportions.

Rather than achieving the additional system definitions via more near field elements, a physics based requirement for pressures and velocities on the patches was examined.

3) The patch adjacency conditions of pressure and velocity, $p_i = p_{i+1}$ and $v_i = v_{i+1}$, provided the physical relationships to achieve both additional defining equations for the system and to assist in forcing a realizable solution. Each

adjacency condition provided a total of (numv-1) * sph_sym equations. For the example, this yielded an additional 40 equations for both pressure and velocity.

Thus, the three components above provided 50, 30, and 80 (40 and 40) defining equations, respectively, for the system. Thus, an over determined system was achieved (since 160 equations now existed to solve for 100 unknowns).

For completeness and further investigation , the MATLAB coding used to define and solve for the sinusoidal signal (corrupted by noise) in this appendix is provided.

```
%function linear_least_squares_fit_to_data_set_SINEWAVE

%clear all
close all

% Create Data Set
max_points = 1024;
time_factor = 1/256;   % sampling period

factor =3 % noise factor - high numbers represent low noise - OPERATOR INPUT
radian_phase = 0; % phase shift of computed data set - OPERATOR INPUT
offset = 0.5; %DC offset of computed data set - OPERATOR INPUT
waveform_magnitude = 1.5; %magnitude of sinewave - OPERATOR INPUT


for integer =1:max_points;
x_value(integer) = integer;
random_number=rand;
magnitude(integer)  =  waveform_magnitude.*sin((integer/256)*2*pi*1+radian_phase)  +
((random_number-0.5)/factor) + offset; %+ 2*sin((4*integer/256)*pi);

end

max_magnitude =sqrt(max(magnitude.^2))
magnitude2 = magnitude;

% Plot Original Computed Data Set including any noise contributions
```

```
figure(1)
plot (x_value*time_factor, magnitude)


% Plot Normalized Data Set
figure(2)
plot (x_value*time_factor, magnitude2)

omega = 2*pi


time =[time_factor:time_factor:max_points*time_factor] ;
data = magnitude2;

% Plot Normalized Data Set (to allow for direct comparison to processed
% data later)
figure(3)
plot(time,data,'b+')
xlabel('time (t)')
ylabel('data value')

time_size = size(time,2)
% Set-up Matrices and Solve for Least Square Coefficients
ones_matrix = ones([ 1 time_size]);

A = [sin(omega*time) ; cos(omega*time); ones_matrix]';

B = data';

A_transpose = A';
```

```
Result_1 = A_transpose * A;

Result_2 = A_transpose * B;

Result_final = Result_1 \ Result_2;



C_value = Result_final(1)
D_value = Result_final(2)
E_value = Result_final(3)

text1 = ['Least Squares Fit Gives:']
text2 = [ num2str(C_value) ' sin(wt) + ' num2str(D_value) ' cos(wt) + ' num2str(E_value)]

position_x = (max(time) + min(time))/8
position_y = (max(data)) % + min(data))/2

% Plot data using least squares derived coefficients
hold on
plot(time,C_value*sin(omega*time) + D_value*cos(omega*time) + E_value, 'r-')

text(position_x, position_y, text2, 'FontSize', 14)
```

APPENDIX C - EXPLANATION OF MATLAB CODING

This appendix supplements Chapter IV – Steps of the Methodology. Detailed explanations of these steps as implemented via the three pieces of MATLAB coding are provided herein.

drake_step1_*.m

This step accomplishes the following:

1) It creates a simulated horizontal transmitting line array of some user-defined length. The elements in the transmitting array are equally spaced and are spaced at 1/6 lambda maximum. Lambda represents the wavelength of the frequency of operation of the array. This transmitting array definition would be bypassed in the event that an actual device was used for acoustic evaluation. In this latter case, the maximum dimension of the acoustic device (its length for a line array) would be entered as input.

2) A near field vertical array is also simulated in his section of the code. This is the array that is used in the near field of the acoustic device under test to acquire response data. In the coding that is provided, the user may define an array of arbitrary length made up of some defined number of equally spaced elements. In actual practice, the data entries here would correspond to the actual construction details of the near field array being used.

3) An estimate of the near field-far field boundary for the defined acoustic device is calculated.

4) The definition of range for the placement of the near field vertical array for the near field measurements is made. A far field range is also defined which places a single receive hydrophone well in the device under test's far field. In the

existing coding the near field range is defined as some percentage of the distance between the enclosing sphere radius (described later) and the near field-far field boundary (near field range = enclosing sphere radius plus factorA*(nf-ff boundary – enclosing sphere radius); the far field range is represented as 100 times this latter factor.

5) Test points corresponding to the (x,y,z) positions for each of the near field array vertical array elements rotated at some defined angular increment about the acoustic device under test are created. These test points form a cylinder about the acoustic device under test. Test points for the far field simulation are also created. In latter case, since the far field will be calculated from the response of a single hydrophone, these test points form a circle about the device under test. It is important to note here that the enclosing sphere symmetry order and the number of angular increments in this section are equal.

In addition to the test points defined here, another set of test points are defined to allow an evaluation of the overall technique when dimensional uncertainty exists. The user has the ability to enter in values of the error associated with the range to the near field array, error in its vertical placement and errors in its angular orientation. In the implementation provided, these test points (with known errors) are associated with simulated data values only.

6) This coding also calculates the complex pressures seen at each of the near field vertical array's elements at each defined test point and at each of the single far field hydrophone's test points. The near field test point response data will be used to simulate actual measurement data; the far field data so computed will be used later to compare to the output from the methodology. The calculations in this step are created by the MATLAB routine "prsvel". This is a custom function call that computes complex response given acoustic parameters and defined source and field points. The complex response at a particular field point is computed as the contribution from each defined source point. The complex

responses so obtained are converted to dB values and normalized for later display. This section would not be executed if actual measurement data existed.

7) The CHIEF enclosing sphere is created within this piece of coding. This spherical surface encloses the acoustic device under test with a known geometrical configuration. This sphere will be diced up into a mosaic of small patches; a single normal pressure and velocity will be assigned to each of these patches at a later point. The radius of this enclosing sphere is deduced from the size of the acoustic device under test and proximity to the location where the near field array will be placed. In like fashion as the placement of the near field array, the coding was implemented to allow a percentage of the separation distance between the maximum radial dimension of the device under test and the location of the near field array to be defined (this is factorB in the coding).

The number of vertical patches and the enclosing sphere's symmetry order are also defined in this section. The number of vertical patches is based upon the assumption that lambda/6 spacing is a reasonable maximum criterion to minimize pressure and velocity changes between adjacent patches (this is examined in more detail below). The symmetry order is based upon the number of vertical patches with the additional constraint that the symmetry should be a highly composite number (a number that may be represented by powers of 2 times 3's and 5's). This latter requirement pertains to the FFT methodology of solving the matrix mathematics and will not be addressed in detail here.

A plot of a single section of the enclosing sphere is made to visualize the enclosing surface.

The lambda/6 criterion of patch spacing was selected based upon an empirical analysis. This analysis looked at the complex response at a point in the far field. This response was formed from the averaged sum of some defined number of common magnitude vectors that were equally spaced (in an arc) but all

had different orientations. The resulting response in the far field was examined for varying vector spacings (ie: varying fractional lambda spacings). This response was compared to that which would be achieved if all vectors were perfectly aligned with the far field point (ie: all vectors pointed in the same direction). This analysis showed that lambda/6 spacing yielded an error of only about 0.4 dB in the far field response. This error was deemed as satisfactory for the intended goals of the technique and was also expected to help minimize the number of patches required for implementation.

The figures below support the above explanation. Three spacings (lambda/6, lambda/12 and lambda/4) are shown in these figures. The corresponding difference in far field magnitude from the first case where all vectors pointed in the same direction are summarized in these figures. In these examples, 50 vectors are shown in each figure.

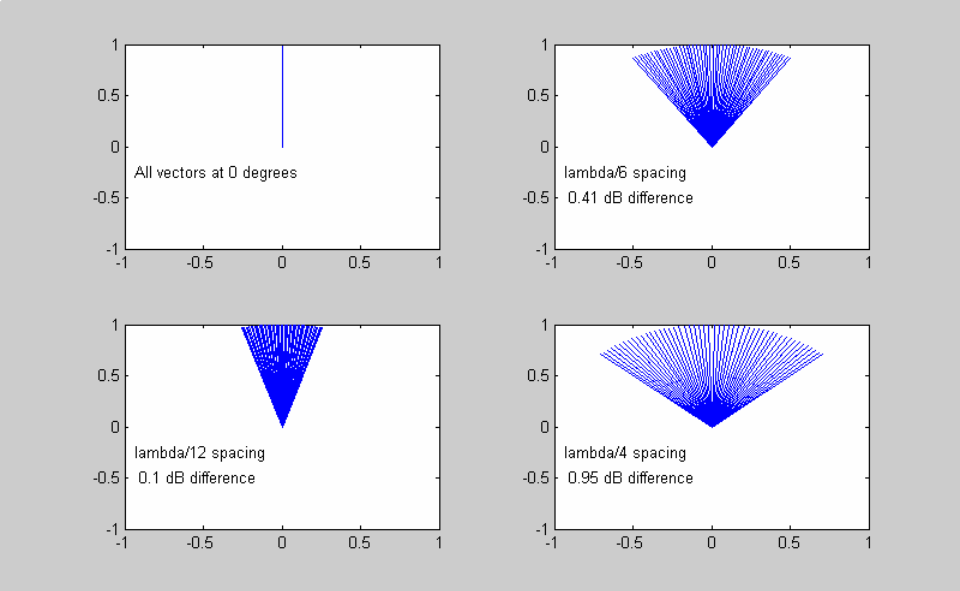


Figure C1 – Vector Spacing Summary and Effect on Far Field Pressure

8) Using CHIEF tools, the coding solves for the **A** and **B** matrices (see CHIEF – Introduction for details on these matrices). Unlike previous versions of

CHIEF, this version allows a cessation of computations after these matrices are solved for.

9) The greens function integral function call is next implemented for all defined near field test points at a single angular orientation along with all enclosing sphere patches (pssph). This function call results in defined values of G and delG for later computations.

10) The Equality matrix is next established for the pressure and velocity adjacency conditions. This matrix consists has dimensions of (sph_numv-1 x sph_sym*sph_numv). This arises because there are sph_numv-1 equality conditions imposed; each condition equates the pressure or normal velocity at a patch to its immediately adjacent patch. This matrix takes the general form of:

```
| 1 -1  0  0  0… 0  0|
| 0  1 -1  0  0… 0  0|
…
| 0  0  0  0  0… 1 -1|
```

The matrix above represents a section (sph_numv-1 x sph_numv) of the overall Equality matrix. The remaining matrix components are all zeros. An Equality matrix for both pressure and velocity adjacency conditions will be provided later.

11) The final operation in this piece of coding establishes a single maximum value for the A matrix and a single maximum value from all computed **G**/rhoc and **delG** values. These values will be used in the next step to normalize the elements of the combined **AB_Matrix** and **M** matrices.

12) The significant data from this step is saved to file.

drake_step2_*.m

This step takes the data obtained from the first step as its input and provides the following functions:

1) The weighting factors alpha0 and beta0 are defined by the user. The alpha0 term is associated with the theoretical Helmholtz Integral relations while the beta0 term is associated with the near field element data. (In practice, this latter data set would be as acquired by the near field array; in this effort the near field data set was simulated.) Since these values can be selected by the user, an emphasis can be placed (when the pressure and velocity values of the enclosing sphere patches are determined) on either the theoretical expectations or on the empirical data obtained. These weighting factors are used along with the values from step 11 in the first piece of coding to create the terms alpha and beta.

2) The augmented $\mathbf{A_a}$ matrix is created by concatenating the **A***alpha, **delG***beta, and Equality (pressure and velocity) matrices; the augmented $\mathbf{B_a}$ matrix is created by concatenating the **B***alpha/rhoc; **G***beta/rhoc and Equality (pressure and velocity) matrices. These two matrices are then combined in such a way to create the **AB_Matrix** of form [A1B1A2B2A3B3…]

3) The augmented $\mathbf{M_a}$ matrix is next created by concatenating zeros (expected outputs for the Helmholtz Integral relations), the near field array element data, and zeros (corresponding to the expected outputs for the $(p_i - p_{i+1})$ and $(v_i - v_{i+1})$ terms).

4) The coding then uses the "rotational_symmetry_solve" function call to create the "pv_solution" matrix. This function call is specific to the CHIEF for MATLAB implementation and allows a problem with rotational symmetry to be solved, **Ax = B**. **A** is a (M x N) matrix, shorthand for MxrN circulant matrix where r is the order of the symmetry; **b** is a (M by 1) matrix of rhs values. The matrix so

obtained, pv_solution, contains individually catered values of pressure and velocity for each patch on the enclosing sphere (as determined by the underlying least squares solution technique contained within the "rotational_symmetry_solve" function call.)

5) It is not just sufficient for the solved for pressures and velocities to yield values (when greens function integrations are used) that agree with those acquired by the near field array elements. Without taking into account the adjacency conditions and the Helmholtz Integral relations, the solved for pressures and velocities could be physically unrealizable. To assess the pv_solution matrix, the function call "rotational_symmetry_multiply" is used. This is another function that is specific to the MATLAB for CHIEF implementation. This function solves a problem with rotational symmetry, **Ax=b**. **A** is a (M x rN) matrix where r is the order of the symmetry and **b** is a (rN by 1) matrix of rhs values. The output from this operation is a new matrix designated as the $\mathbf{M}_{\mathbf{a\text{-}new}}$ matrix.

6) The balance of the coding in this section allows the operator to look at the constituents of both the $\mathbf{M}_{\mathbf{a}}$ and $\mathbf{M}_{\mathbf{a\text{-}new}}$ matrices and assess their agreement. Perhaps the most significant display in this set pertains to the full image outputs of the $\mathbf{M}_{\mathbf{a}}$ and $\mathbf{M}_{\mathbf{a\text{-}new}}$ matrices. These plots contain a lot of information that allow a direct comparison of the two sets of receive array element outputs and allow a comparison in level of the resulting Helmholtz Integral relations and adjacency conditions within the $\mathbf{M}_{\mathbf{a\text{-}new}}$ matrix.

7) The computations in this section can be repeated with new weighting factors (alpha0 and beta0) applied to try and arrive at a best agreement between the two **M** matrices. These computations can be reiterated without the need to re-create the data set from the first step.

8) The primary output of this coding is the pv_solution matrix which will be used to re-establish pressure and velocities on the enclosing sphere for ultimate extrapolation into the far field.

drake_step3_*.m

1) This coding takes the pv_solution matrix from the previous code and splits it into the pressure and velocity components corresponding to each of the enclosing sphere patches. Computations using greens functions integrations are performed to arrive at the **G** and **delG** values using the enclosing sphere geometry and the mid-point element of the near field array field test points and the defined far field point. These values of **G** and **delG** are used along with the newly defiend pressures and velocities to compute pressure at a field point,
p delG + v G.

2) Overplots of the original near field data and that obtained from this technique are displayed. In addition, overplots of the original simulated far field data set and that obtained via this technique are also displayed. In practice, this last overplot would not be made, instead just the computed far field plot would be displayed (since the response would not be known apriori).

```
%step #1 - near field to far field extrapolation technique

close all

V.el=25;V.az=25; % defines reference view for 3-D plots

% factor for placement of near field array: sph_rad +factorA(far_field - sph_rad)
factorA=0.08
% factor for placement of enclosing sphere: height/2 +factorB(range_near_field - height/2)
factorB=0.1305

%define acoustic media parameters and k/frequency values of operation for
%the transmitting array under test
ss=1500;rho=1000;rhoc=ss*rho;
k=20
freq=k*ss/(2*pi)
lambda = ss/freq

%define parameters for vertical near field array
rec_array_spacing = [-3*lambda:lambda/2:3*lambda];  %   vertical spacing in meters of
                                %   receive array elements
max_length_rec_array = max(rec_array_spacing) - min(rec_array_spacing)
num_rec_array_elements = size(rec_array_spacing,2)

%define parameters for horizontal transmitting array (device under test)
height=6*lambda % this is the length of line array in meters
num_array_elements = (round(height/((1/6)*lambda)))+1; %no. of elements comprising array

% calculate far field criteria for defined line array for info purposes
farfield_computed_criteria = (height^2)/(lambda);
max_criteria = farfield_computed_criteria

%define parameters for enclosing sphere based upon wavelength of
%transmitting array and its length
sph_rad = (-(1-factorB)*height/2 -factorA*factorB*max_criteria)/...
   (factorB-(factorA*factorB)-1)

%base number of latitude lines on lambda/6 criterion
sph_numv=ceil(pi*6*sph_rad/(lambda))

%this gives a highly composite number that speeds up FFT process later
sph_sym=2*nearest_small_factor(sph_numv)

%define location of vertical near field array
range_near_field = -(-factorA*max_criteria - sph_rad*(1-factorA))

%define range where far field is guaranteed
range_far_field =100* round((max_criteria)/lambda)*lambda  %simulate far field

% allow operator to assess number of patches and other parameters before
% proceeding
operator_choice=0;
operator_choice = input('Do you want to use these values?  1=yes 0=no ')

if operator_choice == 1 % proceed with computations for a "yes" condition
```

```
%define radian angles for which near field data will be required (the
%number of angles must match the enclosing sphere symmetry)
th=[0:1:sph_sym-1]' *(360/(sph_sym))* pi/180;

%define some degree of positional uncertainty to test methodology
range_error = 0;            % error in meters in range to rec array
rec_array_z_error = 0;      % error in meters for vertical placement of rec array
th_error = 0;               % error in radians of rec array placement

%define all near field point locations
saved = zeros(size(th,1)*num_rec_array_elements,3);

for counter=1:num_rec_array_elements
   test_pts = [range_near_field*sin(th),range_near_field*cos(th),...
         zeros(length(th),1)+rec_array_spacing(counter)];
   test_pts_error =  [(range_near_field+range_error)*sin(th+th_error)...
         ,(range_near_field+range_error)*cos(th+th_error),...
         zeros(length(th),1)+rec_array_spacing(counter)+rec_array_z_error];
   All_nf_test_pts(1+(counter-1)*size(th):counter*size(th),1:3) =test_pts;
   All_nf_test_pts_w_error(1+(counter-1)*size(th):counter*size(th),1:3) ...
      =test_pts_error; %these test points include positional errors

end

%define all far field point locations
test_pts_far=[range_far_field*sin(th),range_far_field*cos(th),zeros(length(th),1)];

% define all source points of the transmitting line array
source_pts =gen_all_xyz_combinations(0,linspace(-height/2,height/2,num_array_elements),0);

%
%surface definition
%create enclosing sphere around array
pssph=patch_define('uv_sphere',[-pi/sph_sym,pi/sph_sym,0,pi],1,1,1,sph_numv,...
   [sph_rad,sph_rad,sph_rad],[0,0,0],[0,0,0]);

% p1sph contains patch data for entire enclosing sphere
p1sph=patches_by_rotation(pssph,sph_sym,'z');
% display single slice of the enclosing sphere
M=plot3_patches(pssph,V);

% define test points for the entire enclosing sphere based upon patch centers
[test_pts_encl_sph,norm_encl_sphere,ar_encl_sphere] = patch_interp(p1sph,.5,.5);

% - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
%simulate pressures at the two defined ranges (near field and far field)

%define normal vector for all defined source points
normals = ones(size(source_pts));
normals(:,1) = 0.*normals(:,1);
normals(:,3) = 0.*normals(:,2);

P_prsvel_near_saved = [];

% compute pressures and velocities for all defined near field test points
```

```
for counter=1:num_rec_array_elements

  near_field_points = All_nf_test_pts_w_error(1+(counter-1)*size(th):...
     counter*size(th),1:3);

  [P_prsvel_near,dummy]=prsvel(source_pts,near_field_points,k,rhoc,normals);
  P_prsvel_near_saved(1+(counter-1)*num_array_elements:counter*...
     num_array_elements,1:size(th))= P_prsvel_near;

end

% compute pressures and velocities for all defined far field test points
[P_prsvel_far,dummy]=prsvel(source_pts,test_pts_far,k,rhoc,normals);


P_prsvel_near_summed_saved = [];

for counter=1:num_rec_array_elements
  summed_value = sum(P_prsvel_near_saved(1+(counter-1)*num_array_elements:...
     counter*num_array_elements,1:size(th,1)));
  P_prsvel_near_summed_saved(counter,:) = summed_value;
end

P_prsvel_far_summed = sum(P_prsvel_far);

All_dBP_reported_near_all = [];

for counter=1:num_rec_array_elements

  entry = P_prsvel_near_summed_saved(counter,:);
  All_dBP_reported_near_all(counter,:) = dB(entry) - dB(max(entry));
  max_rec_array_values(counter,1) = dB(max(entry));
end

dBP_reported_far = db(P_prsvel_far_summed)-db(max(P_prsvel_far_summed));

max_far_field_value = db(max(P_prsvel_far_summed));
% - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%Plot dB pressures (simulated) for all near field test points

for counter=1:num_rec_array_elements
figure
plot(th*180/pi,All_dBP_reported_near_all(counter,:));
axis([0 360 -80 0]);

text2 = ['NEAR FIELD DATA (Rec Ele #' num2str(counter) ') with k = ' num2str(k) ...
      ' ; range = ' num2str(range_near_field) ' meters; array length = ' ...
      num2str(height) ' meters'];
text3 = [' far field criteria =  ' num2str(max_criteria) ' meters; frequency = ' ...
      num2str(freq) ' Hz; lambda = ' num2str(lambda) ' meters' ];

title(text2, 'FontSize', 12)
text(0, -10, text3, 'FontSize', 8)
xlabel('angle (degrees)');
ylabel('pressure in db');
```

```
end

% Plot dB pressures (simulated) for all far field test points
figure
plot(th*180/pi,dBP_reported_far);
axis([0 360 -80 0]);

text2 = ['FAR FIELD DATA with k = ' num2str(k) ' ; range = ' num2str(range_far_field)...
      ' meters; array length = ' num2str(height) ' meters'];

title(text2, 'FontSize', 12);
text(0, -10, text3, 'FontSize', 8);
xlabel('angle (degrees)');
ylabel('pressure in db');

% set up CHIEF parameters to compute A and B matrices
clear O
O.symmetry='ROTATE';
O.symmetry_order={sph_sym,'z'};
O.quadrature_order=2;
O.adaptive_quadrature=adaptive_quadrature_strategy(pssph);
O.rho=rho;
O.c=ss;
O.frequency=freq;
O.matrices_only=1;
[p,v]=solve_chief(pssph,O);

global A_ B_
N=size(A_,1);M=size(A_,2);

% set up to compute greens function integrals for all defined near field
% test points
All_g_nf =[];
All_dg_nf=[];

for counter=1:num_rec_array_elements
   test_pts = All_nf_test_pts(1+(counter-1)*size(th):counter*size(th),1:3);
   [g_nf,dg_nf]=greens_function_integral(pssph,O,test_pts(1,:));

   All_g_nf(1:(sph_sym*sph_numv),counter) = g_nf;
   All_dg_nf(1:(sph_sym*sph_numv),counter) = dg_nf;

end

% Set-up Equality Matrix Structure

E_zeros = zeros(sph_numv-1,sph_numv);
E_equality = zeros(sph_numv-1,sph_numv);

for counter=1:(sph_numv-1)
   E_equality(counter,counter)=1;
   E_equality(counter,counter+1)=-1;
end

E_temp = [E_equality E_zeros];
```

```
for counter = 1:sph_sym-2
   E_temp = [E_temp E_zeros];
end

E_final = E_temp;
E_final_zeros = zeros(size(E_final));

% Determine maximum values in A matrix and in G and delG matrices
factor_A = max(max(abs(A_)))

max_g_nf_temp1= max(abs(All_g_nf(1:(sph_sym*sph_numv),1)));
max_dg_nf_temp1=max(abs(All_dg_nf(1:(sph_sym*sph_numv),1)));

for counter=2:(num_rec_array_elements)

   max_g_nf_temp2 = max(abs(All_g_nf(1:(sph_sym*sph_numv),counter)));

   max_g_nf_temp1 =max(max_g_nf_temp1, max_g_nf_temp2);

   max_dg_nf_temp2= max(abs(All_dg_nf(1:(sph_sym*sph_numv),counter)));
   max_dg_nf_temp1 =max(max_dg_nf_temp1, max_dg_nf_temp2);

end

max_g_nf = abs(max_g_nf_temp1);
max_dg_nf = abs(max_dg_nf_temp1);

% use the following factor to normalize the G and delG data sets (note that
% G also gets divided by rhoc
factor_g_dg_nf = max(max_dg_nf,max_g_nf/rhoc);

%clean up variables that are no longer needed to save memory
clear saved test_ptts source_pts near_field_points dummy summed_value P_prsvel_near_saved...
   P_prsvel_far g_nf dg_nf E_temp entry E_equality P_prsvel_near E_zeros

%save the data set to use in the next step
save drake_full_data_set
end
```

%step #2 - near field to far field extrapolation technique

```
close all
% load data set from step #1
load drake_full_data_set

% set to 0 for first run, set to any other number for subsequent runs of step2 to save decision data
first_run=1;

if first_run==0
   saved_step2_data=[];
else
   load step2_decision_data
end


%for group=1:size(th)  % set group number for data to be displayed - group #1
% is the first set of hydrohone data and its zeros range for group is 1 to size(th)
group =1

% Set alpha0 and beta0 factors (these will be iterated for final solution)

%beta0 places emphasis on the empirical near field data
beta0 = 1000
% alpha0 places emphasis on the theoretical Helmholtz Integral data
alpha0 = 1

% Compute final sclaing factors that will be used
alpha = alpha0/factor_A
beta = beta0/factor_g_dg_nf

%create the augmented A and B matrices
new_A_single_slice = [-A_*alpha;All_dg_nf.'*beta; E_final; E_final_zeros];
new_B_single_slice = [B_*alpha/rhoc;All_g_nf.'*beta/rhoc; E_final_zeros;E_final];

% interleave columns of new_A and new_B single slices into new_AB_single_slice
AB_Matrix = [new_A_single_slice new_B_single_slice];
clear  new_A_single_slice new_B_single_slice

%   now, put indices in order to keep circulant:
%   interleave from [A1A2A3...B1B2B3...] to [A1B1A2B2A3B3...] (block
%   circulant form), where each Am amd Bn is numv by numv in size,
perm_ix=reshape([1:M;M+[1:M]],2*M,1);
AB_Matrix=AB_Matrix(:,perm_ix);


% Set up rhs of equation , Ma, the vector of actual measurements

% Define two vectors of zeros corresponding to A and B (number of rows) and
% the Equality Matrix (number of rows)
Zero0 = zeros(sph_numv,1);
Zero1 = zeros(sph_numv-1,1);

M_temp = [];

% build up M matrix using Zero0, Zero1 and the measured near field
```

```
% pressures at each receive element

for counter =1:size(th);
   % "P_prsvel_near_summed" represents the complex pressure measured at a
   %  particular field point. In this code, it was computed via "prsvel")
   M_temp = [M_temp; Zero0;P_prsvel_near_summed_saved(:,counter); Zero1; Zero1];
end

M_final = M_temp*beta;
clear M_temp

group_size = (3*sph_numv) -2 + size(rec_array_spacing,2);

%solve for pressure and velocity
pv_solution = rotational_symmetry_solve(AB_Matrix, M_final, sph_sym);

%compute Ma-new using solved for pressures and velocities
M_from_pv_soln=rotational_symmetry_multiply(AB_Matrix,pv_solution,sph_sym);

% separate constituents from the two M matrices
M_final_resize1 = reshape(M_final, group_size, sph_sym);
M_final_resize2 = M_final_resize1(sph_numv+1:sph_numv+size(rec_array_spacing,2),:);
M_final_resize3 = reshape(M_final_resize2,(size(rec_array_spacing,2)*sph_sym),1);

M_final_zero0_resize2 = M_final_resize1(1:sph_numv,:);
M_final_zero0_resize3 = reshape(M_final_zero0_resize2,(sph_numv*sph_sym),1);

M_final_zero1_resize2                                                                    =
M_final_resize1(sph_numv+size(rec_array_spacing,2)+1:3*sph_numv+size(rec_array_spacing,2)-2,:);
M_final_zero1_resize3 = reshape(M_final_zero1_resize2,((2*sph_numv-2)*sph_sym),1);

M_from_pv_soln_resize1 = reshape(M_from_pv_soln, group_size, sph_sym);
M_from_pv_soln_resize2                                                                   =
M_from_pv_soln_resize1(sph_numv+1:sph_numv+size(rec_array_spacing,2),:);
M_from_pv_soln_resize3 = reshape(M_from_pv_soln_resize2,(size(rec_array_spacing,2)*sph_sym),1);

M_from_pv_soln_zero0_resize2 = M_from_pv_soln_resize1(1:sph_numv,:);
M_from_pv_soln_zero0_resize3 = reshape(M_from_pv_soln_zero0_resize2,(sph_numv*sph_sym),1);

M_from_pv_soln_zero1_resize2                                                             =
M_from_pv_soln_resize1(sph_numv+size(rec_array_spacing,2)+1:3*sph_numv+size(rec_array_spacing,2)
-2,:);
M_from_pv_soln_zero1_resize3          =          reshape(M_from_pv_soln_zero1_resize2,((2*sph_numv-
2)*sph_sym),1);

max_hydro = 10*(round((max(dB(abs(M_final_resize3))))/10))

ymax = max_hydro;
ymin = max_hydro - 150;

figure
plot(1:size(rec_array_spacing,2)*sph_sym,dB(abs(M_final_resize3)),'k.-',
1:size(rec_array_spacing,2)*sph_sym,dB(abs(M_from_pv_soln_resize3)),'r.-' )

text10 = ['ALL HYDROPHONE DATA (dB); alpha0 = ' num2str(alpha0) ' ; alpha = ' num2str(alpha) ' ;
beta0 = ' num2str(beta0) ' ; beta = ' num2str(beta)];
```

```
title(text10)
axis([1 size(rec_array_spacing,2)*sph_sym ymin ymax])
xlabel('Near Field Array Field Points')
ylabel('Level (dB)')

figure
plot(th*180/pi,min(dB(abs(M_from_pv_soln_resize2))),'r.-' )

text10 = ['min(Ma-new HYDROPHONE DATA) (dB); alpha0 = ' num2str(alpha0) ' ; alpha = ' num2str(alpha) ' ; beta0 = ' num2str(beta0) ' ; beta = ' num2str(beta)];
title(text10)
axis([0 400 ymin ymax])
xlabel('Angle (degrees)')
ylabel('Level (dB)')

M_hydro_difference = dB((M_from_pv_soln_resize3 - M_final_resize3)/max(max(M_final_resize3)));
M_hydro_difference2 = dB(M_from_pv_soln_resize3) - dB(M_final_resize3);

figure
plot(1:size(rec_array_spacing,2)*sph_sym,M_hydro_difference,'g.-');
text10 = ['HYDROPHONE DATA - dB[(computed-actual)/max(actual)] ; alpha0 = ' num2str(alpha0) ' ; beta0 = ' num2str(beta0)];
title(text10)
xlabel('Near Field Array Field Points')
ylabel('Level (dB)')


figure
plot(1:size(rec_array_spacing,2)*sph_sym,M_hydro_difference2,'g.-');
text10 = ['HYDROPHONE DATA - dB[(computed-actual)] ; alpha0 = ' num2str(alpha0) ' ; beta0 = ' num2str(beta0)];
title(text10)
xlabel('Near Field Array Field Points')
ylabel('Level (dB)')


figure
plot(1:sph_numv*sph_sym,dB(abs(M_from_pv_soln_zero0_resize3)),'b.-')
text10 = ['ALL Ma-new Zero0 Data (Helmholtz Integral) (dB); alpha0 = ' num2str(alpha0) ' ; beta0 = ' num2str(beta0)];
title(text10)
axis([1 sph_numv*sph_sym ymin ymax])
ylabel('Level (dB)')
xlabel('Helmholtz Integral Conditions (sphnumv x sphsym)')

figure
plot(1:(2*sph_numv-2)*sph_sym,dB(abs(M_from_pv_soln_zero1_resize3)),'k.-')
text10 = ['ALL Ma-new Zero1 DATA (Pressure/Velocity Equivalence) (dB); alpha0 = ' num2str(alpha0) ' ; beta0 = ' num2str(beta0)];
title(text10)
axis([1 (2*sph_numv-2)*sph_sym ymin ymax])
ylabel('Level (dB)')
xlabel('Pressure/Velocity Adjacencies ((2*sphnumv-2) x sphsym)')

figure
plot(th*180/pi,max(dB(abs(M_from_pv_soln_zero0_resize2))),'b.-')
```

```
text10 = ['max(Ma-new Zero0 Data)- (Helmholtz Integral) (dB); alpha0 = ' num2str(alpha0) ' ; beta0 = '
num2str(beta0)];
title(text10)
axis([0 400 ymin ymax])
xlabel('Angle (degrees)')
ylabel('Level (dB)')

figure
plot(th*180/pi,max(dB(abs(M_from_pv_soln_zero1_resize2))),'k.-')
text10 = ['max(Ma-new Zero1 DATA) - (Pressure/Velocity Equivalence) (dB); alpha0 = ' num2str(alpha0) '
; beta0 = ' num2str(beta0)];
title(text10)
axis([0 400 ymin ymax])
xlabel('Angle (degrees)')
ylabel('Level (dB)')

M_hydro_zero0_diff = dB(min(M_from_pv_soln_resize2)) - dB(max(M_from_pv_soln_zero0_resize2));
M_hydro_zero1_diff = dB(min(M_from_pv_soln_resize2)) - dB(max(M_from_pv_soln_zero1_resize2));

figure
plot(th*180/pi,M_hydro_zero0_diff,'g.-');
text10 = ['Ma-new [ min(HYDROPHONE) - max(Zero0) ] (dB) ; alpha0 = ' num2str(alpha0) ' ; beta0 = '
num2str(beta0)];
title(text10)
xlabel('Angle (degrees)')
ylabel('Level (dB)')

figure
plot(th*180/pi,M_hydro_zero1_diff,'g.-');
text10 = ['Ma-new [ min(HYDROPHONE) - max(Zero1) ] (dB) ; alpha0 = ' num2str(alpha0) ' ; beta0 = '
num2str(beta0)];
title(text10)
xlabel('Angle (degrees)')
ylabel('Level (dB)')


figure
plot(th*180/pi,min(dB(abs(M_from_pv_soln_resize2))),'r.-',...
   th*180/pi,max(dB(abs(M_from_pv_soln_zero0_resize2))),'b.-',...
   th*180/pi,max(dB(abs(M_from_pv_soln_zero1_resize2))),'k.-')
text10 = ['Ma-new - min(HYDROPHONE), max(Zero0), max(Zero1); alpha0 = ' num2str(alpha0) ' ; alpha
= ' num2str(alpha) ' ; beta0 = ' num2str(beta0) ' ; beta = ' num2str(beta)];
text11 = ['   red: min(hydrophone); blue: max(Zero0); black: max(Zero1)'];

title(text10)
axis([0 400 ymin ymax])
xlabel('Angle (degrees)')
ylabel('Level (dB)')
text(0, ymin+10, text11, 'FontSize', 8)

clim =[ymin,ymax]
figure
subplot(1,2,1);imagesc(max(-200,db(M_from_pv_soln_resize1)),clim)
colorbar
title('Ma-new - from pv solution')
```

```
subplot(1,2,2);imagesc(max(-200,db(M_final_resize1)),clim)
colorbar
title('Ma - original')

%clean up variables that are no longer needed
clear A_ B_ All_dg_nf All_g_nf E_final E_final_zeros M_final M_from_pv_soln


% compute parameters to be saved and examined later

saved1 = min(min(dB(abs(M_from_pv_soln_resize2)))) %minimum of the new hydrophone data minimum set
saved2 = max(min(dB(abs(M_from_pv_soln_resize2)))) % maximum of the new hydrophone data minimum set
saved3 = mean(min(dB(abs(M_from_pv_soln_resize2)))) %average of the new hydrophone data minimum set

saved4 = min(max(dB(abs(M_from_pv_soln_zero0_resize2)))) %minimum of the new zero0 data maximized set
saved5 = max(max(dB(abs(M_from_pv_soln_zero0_resize2)))) %maximum of the new zero0 data maximized set
saved6 = mean(max(dB(abs(M_from_pv_soln_zero0_resize2)))) %average of the new zero0 data maximized set

saved7 = min(max(dB(abs(M_from_pv_soln_zero1_resize2)))) %minimum of the new zero1 data maximized set
saved8 = max(max(dB(abs(M_from_pv_soln_zero1_resize2)))) %maximum of the new zero1 data maximized set
saved9 = mean(max(dB(abs(M_from_pv_soln_zero1_resize2)))) %average of the new zero1 data maximized set

saved10 = min(M_hydro_difference2) %minimum of the (new-original) hydrophone data set
saved11 = max(M_hydro_difference2) %maximum of the (new-original) hydrophone data set
saved12 = mean(M_hydro_difference2) %average of the (new-original) hydrophone data set

saved13 = min(M_hydro_zero0_diff); %minimum of the (hydrophone(new) - zero0(new) data
saved14 = max(M_hydro_zero0_diff); %maximum of the (hydrophone(new) - zero0(new) data

saved15 = min(M_hydro_zero1_diff); %minimum of the (hydrophone(new) - zero1(new) data
saved16 = max(M_hydro_zero1_diff); %maximum of the (hydrophone(new) - zero1(new) data


% order of data -

% alpha0 (1)
% beta0  (2)

% min_hydro_diff  (3)
% max_hydro_diff  (4)
% mean_hydro_diff (5)

% min_hydro_minus_zero0 (6)
% max_hydro_minus_zero0 (7)
% min_hydro_minus_zero1 (8)
% max_hydro_minus_zero1 (9)
```

```
% min_new_hydro_set (10)
% max_new_hydro_set (11)
% mean_new_hydro_set (12)

% min_new_zero0_set (13)
% max_new_zero0_set (14)
% mean_new_zero0_set(15)

% min_new_zero1_set  (16)
% max_new_zero1_set  (17)
% mean_new_zero1_set (18)

step2_data = [alpha0 beta0 saved10 saved11 saved12 saved13 saved14 ...
      saved15 saved16 saved1 saved2 saved3 saved4 saved5 saved6 ...
      saved7 saved8 saved9];

if first_run ==0
saved_step2_data = [step2_data]
else
saved_step2_data = [saved_step2_data; step2_data];
end

header_info={'alpha0 (1)','beta0 (2) ','min_hydro_diff (3)', 'max_hydro_diff (4)','mean_hydro_diff (5)',...
         'min_hyd_-_zero0 (6)',' max_hyd_-_zero0 (7)','min_hyd_-_zero1 (8)' 'max_hyd_-_zero1 (9)'}

header=(1:9);
[header; saved_step2_data(:, 1:9)]


save step2_decision_data saved_step2_data;

wk1write('step2_data_wk1_file', saved_step2_data)

save drake_full_data_set2
```

```
%step #3 - near field to far field extrapolation technique

close all

%load data set from step #2
load drake_full_data_set2


% pull pressure and velocity values from previous step for use in
% green's function integrations

pv_soln1= reshape(pv_solution,2,size(pv_solution,1)/2);

p_encl_sphere = pv_soln1(1,:).';
v_encl_sphere = pv_soln1(2,:).'/rhoc;

clear pv_soln1 pv_solution


%select near field array element in the middle
counter = ceil(num_rec_array_elements/2)

%evaluate green's function from all patches to all near field points and
%calculate pressures for all test points for the middle recieve array element

test_pts = All_nf_test_pts(1+(counter-1)*size(th,1):counter*size(th,1),1:3);

[g_nf,dg_nf]=greens_function_integral(pssph,O,test_pts);
fp_temp=calculate_field_pressures(dg_nf,g_nf,p_encl_sphere,v_encl_sphere);

max_field_point = max(fp_temp);
Field_pressures(counter,1:size(th,1)) = dB(fp_temp) -  max_rec_array_values(counter,1);

clear fp_temp g_nf dg_nf

%Plot dB Pressures for near field (original and that obtained via
%technique)

figure
plot(th*180/pi,All_dBP_reported_near_all(counter,1:size(th,1)),'b.-', ...
   th*180/pi,Field_pressures(counter,1:size(th,1)), 'r.:');

axis([0 360 -80 0]);

text2 = ['NEAR FIELD DATA (Rec Ele #' num2str(counter) ') with k = ' ...
      num2str(k) ' ; range = ' num2str(range_near_field) ' meters; array length = ' ...
      num2str(height) ' meters'];
text3 = ['  far field criteria =  ' num2str(max_criteria) ' meters; frequency = ' ...
      num2str(freq) ' Hz; lambda = ' num2str(lambda) ' meters' ];
text4 = ['  alpha = ' num2str(alpha) ' ; beta = ' num2str(beta) ' ; alpha0 = ' ...
      num2str(alpha0) ' ; beta0 = ' num2str(beta0) ]
text6 = ['  number of equally spaced receive array elements: ' num2str(num_rec_array_elements)...
      ' ; its length is ' num2str(max_length_rec_array) ' meters'];
text5 = ['  red (computed via method) ; blue (measured/simulated)'];
text7 = ['  enclosing sphere radius = ' num2str(sph_rad) ' meters; sph numv = ' ...
      num2str(sph_numv) ' ; sph sym = ' num2str(sph_sym)];
```

```
title(text2, 'FontSize', 12)
text(0, -10, text3, 'FontSize', 8)
text(0, -70, text4, 'FontSize', 8)
text(0, -60, text5, 'FontSize', 8)
text(0, -65, text6, 'FontSize', 8)
text(0, -75, text7, 'FontSize', 8)

xlabel('angle (degrees)');
ylabel('pressure in db');



%Plot dB Pressures differences for near field (original and that obtained via
%technique)

figure
plot(th*180/pi,All_dBP_reported_near_all(counter,1:size(th,1))- Field_pressures(counter,1:size(th,1)),'g.-');

text2 = ['NF DATA Difference (Rec Ele #' num2str(counter) ') with k = ' ...
      num2str(k) ' ; range = ' num2str(range_near_field) ' meters'];

title(text2, 'FontSize', 12)

xlabel('angle (degrees)');
ylabel('db');



%evaluate green's function and calculate pressure for all far field points
[g_ff,dg_ff]=greens_function_integral(pssph,O,test_pts_far);
P_ff_gf1=calculate_field_pressures(dg_ff,g_ff,p_encl_sphere,v_encl_sphere);

%normalize far field pressures computed via Greens Function and convert to dB
%P_est_reported_ff1 = db(P_ff_gf1)-db(max(P_ff_gf1));
P_est_reported_ff1 = db(P_ff_gf1)-max_far_field_value;

clear g_ff dg_ff


%Plot dB Pressures for far field (original and that obtained via
%technique)
figure
plot(th*180/pi,dBP_reported_far,'b.-',th*180/pi,P_est_reported_ff1, 'r.:' );
axis([0 360 -80 0]);

text2 = ['FAR FIELD DATA (on axis) with k = ' num2str(k) ' ; range = ' ...
      num2str(range_far_field) ' meters; array length = ' num2str(height) ' meters'];
text3 = ['  far field criteria =  ' num2str(max_criteria) ' meters; frequency = ' ...
      num2str(freq) ' Hz; lambda = ' num2str(lambda) ' meters' ];
text4 = ['  alpha = ' num2str(alpha) ' ; beta = ' num2str(beta) ' ; alpha0 = ' ...
      num2str(alpha0) ' ; beta0 = ' num2str(beta0) ]
text5 = ['  red (computed via method) ; blue (measured/simulated)'];
text6 = ['  number of equally spaced receive array elements: ' num2str(num_rec_array_elements)...
      ' ; its length is ' num2str(max_length_rec_array) ' meters'];
text7 = ['  enclosing sphere radius = ' num2str(sph_rad) ' meters; sph numv = ' ...
```

```
      num2str(sph_numv) ' ; sph sym = ' num2str(sph_sym)];

title(text2, 'FontSize', 12)
text(0, -10, text3, 'FontSize', 8)
text(0, -70, text4, 'FontSize', 8)
text(0, -60, text5, 'FontSize', 8)
text(0, -65, text6, 'FontSize', 8)
text(0, -75, text7, 'FontSize', 8)

xlabel('angle (degrees)');
ylabel('pressure in db');


%Plot dB Pressures differences for far field (original and that obtained via
%technique)
figure
plot(th*180/pi,dBP_reported_far-P_est_reported_ff1, 'g.-' );

text2 = ['FF DATA Difference(on axis) with k = ' num2str(k) ' ; range = ' ...
      num2str(range_far_field) ' meters'];

title(text2, 'FontSize', 12)

xlabel('angle (degrees)');
ylabel('db');




% Re-plot All Hydrophone Data that corresponds to the
% data set presented.

figure
plot(1:size(rec_array_spacing,2)*sph_sym,abs(M_final_resize3), ...
   1:size(rec_array_spacing,2)*sph_sym,abs(M_from_pv_soln_resize3) )

text10 = ['ALL HYDROPHONE DATA ; alpha0 = ' num2str(alpha0) ' ; alpha = ' ...
      num2str(alpha) ' ; beta0 = ' num2str(beta0) ' ; beta = ' num2str(beta)];
title(text10)


clim =[ymin,ymax]
figure
subplot(1,2,1);imagesc(max(-200,db(M_from_pv_soln_resize1)),clim)
colorbar
title('Ma-new - from pv solution')

subplot(1,2,2);imagesc(max(-200,db(M_final_resize1)),clim)
colorbar
title('Ma - original')

range_error % error in meters
rec_array_z_error  % error in meters
th_error  % error in radians

save drake_full_data_set_final
```

LIST OF REFERENCES

[1] Naval Undersea Warfare Center Division Newport - Code 216 "Underwater Sound Reference Division", Brochure Set, 1998.

[2] Forsythe, S.E., "CHIEF for MATLAB User's Guide", Naval Undersea Warfare Center Division Newport, internal document, pp 1- 5, 2000.

[3] Bobber, R.J., "Underwater Electroacoustic Measurements", U.S. Government Printing Office, pp 122, 1970.

[4] Ziomek, L.J., "Fundamentals of Acoustic Field Theory and Space-Time Signal Processing", CRC Press, pp 405-413, 1995.

[5] Lastinger, J.L., "Acoustic Measurements", Naval Research Laboratory, internal document, pp 41, 1985.

[6] Benthien G.W., Barach D., Gillette D. "CHIEF (Combined Helmholtz Integral Equation Formulation) Users Manual, Naval Ocean Systems Center, San Diego, pp 1-2, 1988.

[7] Strang, G., "Introduction to Applied Mathematics", Wellesley-Cambridge Press, pp 35-39, 138-140, and 297,1986.

[8] Kreyszig E., "Advanced Engineering Mathematics", John Wiley & Sons, pp 512-580, 1988.

[9] "Acoustic measurements for Transducer Repair Facility (TRF) Sonar Test Personnel", Naval Underwater Systems Center, New London, CT, internal document, pp 80-83, 1988.

[10] Kinsler, L.E., Frey, A.R., Coppens, A.B., Sanders,J.V. "Fundamentals of Acoustics", John Wiley & Sons, pp 176-179, and 191, 2000.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, VA

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, CA

3. Dr. Steve Baker
   Naval Postgraduate School
   Monterey, CA

4. Mr. Stephen Forsythe
   Naval Undersea Warfare Center Division Newport
   Newport, RI

5. Dr. Joseph Zalesak
   Naval Undersea Warfare Center Division Newport
   Newport, RI

6. Dr. A.L. Van Buren
   Naval Undersea Warfare Center Division Newport
   Newport, RI

7. Technical Library
   Naval Undersea Warfare Center Division Newport
   Newport,RI